DOCUMENT RESUME

ED 239 306 CS 208 141

TITLE	Computer Utilization in Composition Instruction Specifications. Final Report.
INSTITUTION	Southwest Regional Laboratory for Educational Research and Development, Los Alamitos, Calif.
SPONS AGENCY	National Inst. of Education (ED), Washington, DC.
REPORT NO	SWRL-TN-2-80/06; SWRL-TN-2-82/37; SWRL-WP-2-82/02-04
PUB DATE	Nov 82
NOTE	682p.; Several pages marginally legible. Cooperative Inquiry on Composition Instruction Deliverable 8a. Incorporates several previously published SWRL Technical Notes. For a related document, see CS 208142.
PUB TYPE	Guides - Classroom Use - Guides (For Teachers) (052)
EDRS PRICE	MF04/PC28 Plus Postage.
DESCRIPTORS	*Computer Assisted Instruction; *Computer Programs; Editing; Elementary Education; Guidelines; Microcomputers; Prewriting; Program Guides; Revision (Written Composition); Word Processing; *Writing Exercises; *Writing Instruction; *Writing Skills

ABSTRACT

Divided into two sections, this report outlines several components of computer based instruction in the composing process. The first section, which focuses on the use of computers in editing, generating ideas, and producing text, presents a users' manual for composition instruction on a microcomputer, a general description of how computer based practice in editing might operate, an outline of heuristics used to generate ideas for descriptive writing, and specifications for interactive computer based composing of narratives. It includes samples of how such programs might accommodate specific content. The second section focuses on the use of computers for drill and practice in basic mechanical skills. After outlining this supplementary instruction, the section provides a sample program reinforcing capitalization and punctuation skills through practice in addressing envelopes. The report concludes with extensive appendixes, including (1) descriptions of other computer instruction programs, (2) reports on professional meetings, (3) a book review, (4) courseware reviews, (5) a discussion of microcomputer systems, (6) a proposal for an elementary school composition progam, and (7) a copy of a discussion of computers, word processors, and composition instruction printed in a college publication. (MM)

ED239306

SWRL EDUCATIONAL RESEARCH AND DEVELOPMENT
4665 LAMPSON AVENUE
LOS ALAMITOS, CALIFORNIA · 90720

COMPUTER UTILIZATION IN COMPOSITION INSTRUCTION SPECIFICATIONS FINAL REPORT

November 1982

COOPERATIVE INQUIRY ON COMPOSITION INSTRUCTION
Deliverable 8.a

CS 208 141

COMPUTER UTILIZATION IN COMPOSITION INSTRUCTION

SPECIFICATIONS FINAL REPORT

November 1982

SWRL Educational Research and Development
4665 Lampson Avenue
Los Alamitos, CA 90720

INTRODUCTION

Under a contract with the National Institute of Education, SWRL Educational Research and Development has been investigating the use of microcomputers in composition instruction. This report presents general specifications for computer-based composition instruction.

During this work, a number of tasks have been undertaken to further staff's understanding of the role of computers in instruction and in composition instruction in particular. Computer instruction in nearby school districts was surveyed (see Appendix A). Professional meetings were attended to hear reports of current research into educational computing (see Appendix B). Books, articles, and reports of computer-based instruction were read (see Appendix C for a review of one such book). Some currently available computer courseware for education was reviewed (see examples in Appendix D). A two-day conference was held to bring together people currently using computers in composition instruction (reported in Joseph Lawlor, Ed., Computers in composition instruction, SWRL, 1982). In addition, computer hardware and software configurations were analyzed for their appropriateness for computer-based composition instruction and for research (see Appendix E). All of this work has furthered SWRL's knowledge about computer instruction.

Actual work in computer-based composition instruction has moved in two complementary directions. The first direction of inquiry has emphasized the composing process and sophisticated computer-student-text interactions. (See Appendix G for a discussion of the role of computers in teaching the composing process.) This work has focused on four instructional areas: sentence combining, editing, generating ideas, and producing text.

The work on sentence combining has been carried further than work in the other areas and is documented in a separate report. Work on the other three areas constitutes Part I of the present report.

While this inquiry has focused on higher-level processes of composing, work has also been undertaken on more mechanical aspects of composition instruction. Computers offer great possibilities for instruction in complex processes, but they are also of considerable value for instruction in more mundane, more mechanical aspects of composition. Although some theoretists have decried drill-and-practice instruction on the computer, such instruction has many values:

1. It permits more individualization of instruction.
2. It can relate diagnosis and instruction.
3. It provides students with immediate feedback.
4. It can be more interesting to students than comparable workbook instruction and practice.
5. It frees teachers so that they may devote their time to working with students in the actual composing process.

Consequently, the specifications found in Part II focus on the use of computers for drill-and-practice with basic mechanical skills. This computer-based instruction is treated as supplementary to a full program of composition instruction.

Ann Humes had primary responsibility for this project and did the major work. Bruce Cronnell also worked on parts of the project, as did Joseph Lawlor and Larry Gentry. (Cronnell was primarily responsible for putting together this report.) Jerry Bailey did the computer programming reported in Part II.

COMPUTER UTILIZATION IN COMPOSITION INSTRUCTION

SPECIFICATIONS FINAL REPORT

Table of Contents

PART I: THE COMPOSING PROCESS

PART I

SPECIFYING COMPUTER-BASED INSTRUCTION IN THE COMPOSING PROCESS

Introduction

A. User Manual: Composition Instruction on a Microcomputer

B. Computer-Based Practice in Editing

C. Computer Instruction on Generating Ideas for Writing Description

D. Computer Instruction for Generating and Revising/Editing Narrative Text

PART I

SPECIFYING COMPUTER-BASED INSTRUCTION IN THE COMPOSING PROCESS

Introduction

A complete curriculum of instruction in the composing process would have many components, not all of which would utilize computers--because much of learning how to write depends on teacher-student and student-student interactions. (For description of a complete composition curriculum, see Ann Humes, TN 2-80/21, Specifications for composition instruction.) This part of the report outlines several components of computer-based instruction in the composing process.

A. User Manual: Composition Instruction on a Microcomputer

This User Manual describes a basic program in computer-based instruction in the composing process. Three of the components are discussed further in the following three sections; the fourth component--sentence combining--is treated in greater detail in a separate report.

B. Computer-Based Practice in Editing

Editing--the correcting of mechanical errors in writing--is an important aspect of composing a final written product. Editing is of particular interest in computer-based composition instruction because editing and revising are much easier with word processors than with paper and pencil; consequently, students may be more willing to edit and revise. This section consists of a general description of how computer-based practice in editing might operate, followed by sample specifications for actual computer programming.

C. Computer Instruction on Generating Ideas for Writing Description

Students frequently have difficulty in generating ideas to use in writing. This section outlines computer-based instruction that teaches heuristics for generating ideas for one kind of writing--description. The general specifications are followed by detailed specifications for programming, as well as sample screen displays that illustrate how the programmed instruction might appear on a computer terminal.

D. Computer Instruction for Generating and Revising/Editing Narrative Text

A goal of computer-based composition instruction is to have students actually compose compositions on a computer that has word-processing capabilities. This section suggests how such interactive computer-based composing might operate for the writing of narratives. The specifications outlined here cover not only the drafting of a text, but also the editing and revising of that text. These specifications are followed by a sample of how such specifications might be programmed with specific content.

User Manual

USER MANUAL

COMPOSITION INSTRUCTION ON A MICROCOMPUTER

Ann Humes

SWRL Educational Research and Development
4665 Lampson Avenue
Los Alamitos, CA 90720

USER MANUAL

COMPOSITION INSTRUCTION ON A MICROCOMPUTER

BACKGROUND INFORMATION AND RATIONALE

Scholarly concern with composition has traditionally focused on analysis and criticism of the resulting document rather than on the means of producing that document. However, authorities have become increasingly concerned with instruction that stresses the process rather than the product of writing (Humes, 1980b). Correspondingly, SWRL composition instruction, designed for use on a microcomputer, focuses on the composing process and is based on an instructional model of that process (Humes, 1980a).

The microcomputer is one of the newest and most versatile media for human communication. Historically, computer-assisted instruction has been used to teach math, reading, and component skills of writing, such as spelling and punctuation. Hardware and software limitations precluded the design of computer instruction that required actual composing activities. Only relatively recent developments, such as the word processor, have enabled designers and developers of instruction to produce courseware that involves the student in actual text production and manipulation.

In the instructional application described here, the microcomputer functions as both recipient and "speaker" of the communication. As a recipient, the computer processes the written input and requires that certain conventions be followed for the communication to be received and deciphered accurately. For writing instruction, this rigidity can teach students that certain rules, such as the use of periods to end sentences, are necessities that often determine whether a communication is received at all or, if it is received, whether the message is understood accurately.

The microcomputer also functions as a speaker because it interacts with the student by presenting instruction and examples, providing interesting practice, pointing out errors, and requesting clarification of ambiguities. This interactive feedback makes the computer an ideal audience for the student writer. For example, microcomputer-presented exercises guide and are guided by the student's responses. Therefore, the student can work in privacy, proceeding at his or her own rate. This privacy can make instructional tasks less threatening.

Microcomputer instruction offers other benefits to students as well. For example, it permits truly individualized composition instruction because students proceed at their own pace and are given as little or as much instruction as they need, as indicated by their computer-analyzed performance.

Immediate feedback is another microcomputer capability that is beneficial to students. A time delay is commonly necessary for teacher response to a student's writing. However, the computer is capable of an instantaneous response, thus providing the student with immediate reinforcement or assistance.

Microcomputer instruction is also highly motivating to students because of its game-like nature. Additionally motivating are the lively graphics: They not only add interest, but they also facilitate presentation of instruction that is less abstract. For example, in the sentence-combining module, students do not merely see a set of simple sentences and the resulting combined sentence frozen on the page along with an abstract explanation of the process; rather, instruction on the computer moves the words and cues around so that students actually see how to combine sentences--the graphics reinforce the instructional concepts.

Flexibility is another asset of microcomputers with word-processing capabilities. Part of the problem for the student writer is the fixed, immutable quality of the symbols on the page: The process of changing, making insertions, and rearranging the text is so laborious that the writer frequently gives up long before he/she is satisfied. However, word-processing features enable the writer to make changes with ease. The writer can add, delete, or change single characters, words, phrases, sentences, and paragraphs in order to improve the final product.

PREREQUISITES

Student Prerequisites

Students must be in grades 6-12 and have previous experience with the microcomputer that will be used in the school. They must also have previous experience with typing because SWRL materials do not provide instruction on this skill.

Students must be free to work with on-line instruction for 20 minutes per day, the approximate amount of time for which the microcomputer lessons are designed. Individual students may complete lessons in less time; however, if the student finishes a lesson within 10 minutes or less after logging on the microcomputer, the next lesson in sequence is presented.

Hardware Prerequisites

In order to implement the instructional materials, a school will need the following equipment, as described in detail by Russell (1980):

Apple II microcomputer (keyboard, CPU, power supply, and CRT).

At least one 5 1/2-inch floppy disk drive.

An interval/frequency timer (e.g., CCS clock card for Apple II).

A printer (optional).

CONTENT OF INSTRUCTION

The content of composition instruction on the microcomputer consists of sentence-combining techniques, revision processes and strategies, and actual text production.

Sentence Combining

Sentence-combining instruction and practice enhance students' syntactic fluency (Lawlor, 1980; Humes, 1980c). Therefore, sentence-combining techniques should be part of any program that teaches the composing process. Furthermore, research indicates that this instruction can be appropriately implemented at this level (Lawlor, 1980; Humes, 1980a). Comprehensive content for sentence-combining instruction is described in Lawlor (1981).

Revision

Changing the content or arrangement of composing occurs when the writer reviews the text and sees mismatches between his/her intended solution of the composing problem and the actual resolution incorporated in the text. Therefore, writers need to learn that revision is a normal and expected part of writing. Despite this need, the process of revision is largely neglected in current composition instruction. However, the computer can make text revision relatively painless and even enjoyable. Consequently, the student will learn that the writing process takes place over time, resulting in a written product that is gradually shaped into a form acceptable to both writer and audience (see Gentry, 1980). The knowledge, skills, and techniques for changing, or revising, text are discussed by Gentry (1980) and described by Humes (1980c).

Text Production

Text production includes generating, arranging, and translating content. Generating entails gathering information to write, whether that information is material from external sources of content discovered within the writer's mind. When data are not gathered from external sources, the writer must search his/her own mind for information. This search may involve the use of heuristic probes for generating content; these probes generally are formalized sets of self-questions the writer (or, in this case, the microcomputer) uses to draw on content stored in memory.

Arranging is the process of ordering content; it contributes form to a final product. Arranging may involve "deletion" of content when more content is generated than is needed. Students learn both the process of arranging and specific paradigms that can facilitate ordering content for different discourse modes and product forms.

Translating is the process of transforming content from one form of symbolization (thought) into another form of symbolization (graphic representation). Translating while functioning simultaneously in other elements of the process makes huge demands on the writer's cognitive processes (Humes, 1980a). This mental load becomes less difficult as an increasing number of translating skills become automatic rather than consciously driven. This move to "automation" is expedited by instruction and practice. Translating skills can be categorized as skills of language, coherence, and form.

Appropriate content for composition instruction on text production is described by Humes (1980c).

INSTRUCTIONAL COURSEWARE

The instructional courseware is designed to be used by students in grades 6-12 and is to be completed on a microcomputer. These materials do not consist of the typical workbook-like multiple-choice and fill-in materials characteristic of currently available computerized instruction. Rather, these interactive materials are designed to involve students directly in productive tasks. For example, in the lessons on sentence combining, students actually construct combined sentences rather than select the appropriately combined answer choice.

References

Gentry, L. A. Textual revision: A review of the research. (Technical Note 2-80/11.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980. (ERIC No. ED 192 355)

Humes, A. An instructional model of the composing process. (Technical Note 2-80/10.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980a. (ERIC No. ED 192 379)

Humes, A. The composing process: A review of the literature. (Technical Note 2-80/09.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980b. (ERIC No. ED 192 378)

Humes, A. Specifications for composition instruction. (Technical Note 2-80/21.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980c. (ERIC No. ED 192 377)

Lawlor, J. Improving student writing through sentence combining: A literature review. (Technical Note 2-80/05.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980. (ERIC No. ED 192 356)

Lawlor, J. Instructional specifications for sentence combining. (Technical Note 2-81/08.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1981. (ERIC No. ED 210 701)

Russell, W. Microcomputer systems: A tool for instruction. (Technical Note 2-80/13.) Los Alamitos, Calif.: SWRL Educational Research and Development, 1980.

Editing

SOUTHWEST REGIONAL LABORATORY
WORKING PAPER

DATE: September 30, 1982

NO: WP 2-82/04

COMPUTER-BASED PRACTICE IN EDITING

Bruce Cronnell

ABSTRACT

General specifications are provided for computer-based practice in editing text mechanics (capitalization, punctuation, spelling, usage). The student works with 2-4 mechanics rules at a time. For each set of rules, three kinds of practice are provided: (1) choice (between correct and incorrect sentences); (2) correction (of incorrect sentences); (3) dictation (i.e., typing dictated sentences, with computer analysis of student input).

COMPUTER-BASED PRACTICE IN EDITING

Bruce Cronnell

One goal of computer-based instruction in writing is to help students edit their compositions, particularly those compositions written on a word processor. This paper outlines possible procedures for teaching students to edit for mechanics errors in their writing. The procedures also provide students with a review of mechanics--and could even be utilized for practice when students are receiving initial instruction on mechanics.

A complete editing program would contain the full set of mechanics rules--capitalization, punctuation, spelling, usage--appropriate for the grade level of the intended users.* The specific rules that students would practice could be decided in three ways:

1. student choice: the student would choose from a menu those rules (or that rule) she/he wished to practice.
2. teacher choice: the teacher would choose those rules (that rule) a student or a group of students would practice. Teacher choice could be implemented in two ways. First, the teacher could tell students to select specific rules from a menu. Secondly, the teacher could enter students' names and the rules to be practiced, and the computer would automatically place students on the assigned rules.

*For convenience, this paper is illustrated with examples of low-level capitalization instruction.

3. by pretest: the computer would assign students to the appropriate rules on the basis of pretest scores. The pretest would be of the form described in the "Correction" section (below) and would begin with the simplest rules, moving to more difficult rules as students displayed proficiency on the easier ones. The computer would automatically assign practice on those rules for which students performed below criterion.

Although these three procedures could permit practice with only one rule, practice would be more effective if 2-4 rules were covered at the same time. With several rules, students cannot simply look for a specific structure, but rather must discriminate among various structures. For the pretest option, the computer would continue testing until 2-4 rules were found for which students performed below criterion. For the student-choice and teacher-choice options, the number of rules to be practiced could be controlled in two ways:

1. The program could "insist" that the student or teacher choose 2-4 rules.
2. The program could permit the student or teacher to choose only one rule, but if only one rule were chosen, the computer would automatically assign 1-3 additional rules for practice (rules that were at a similar difficulty level to the one chosen).

After content for practice is chosen/assigned, the student is given the option of reviewing the rules to be practiced. Following this optional rule review, students receive three kinds of practice: choice, correction, dictation.

Choice

The "Choice" section provides basic practice in discriminating the correct application of the rules under study. Two sentences are presented, one with correct rule application, one without correct rule application; e.g.:

Which sentence is correctly capitalized?

a. Where is my friend's car?

b. where is my friend's car?

The student types the letter of the correct sentence.

If the student types the correct letter, another item is presented. If the student types the wrong letter, the computer presents the rule that should have applied (for the above example, "The first word in a sentence is always capitalized"). Then the incorrect sentence is displayed, with the error graphically highlighted (for the above example, "w"). Then the error is corrected by the computer (e.g., "w" changes to "W") and a comment is displayed (e.g., "Since <u>Where</u> is the first word in the sentence, it must be capitalized"). Then another item is presented.

New items are presented until the student has responded, for each rule, with either three correct answers or three incorrect answers. (Once the student has responded either correctly to three items for a rule or incorrectly to three items for a rule, no additional items for that rule are presented.) When the student has responded correctly to three items for each of the rules being studied, she/he moves to the "Correction" section. If the student has responded incorrectly to three items for one or more rules, additional practice is provided for each rule with three incorrect items.

After another example of the rule is presented, additional practice consists of more sentence pairs to choose between. If the student now reaches criterion on all rules receiving additional practice, she/he moves to the "Correction" section. If the student still does not meet the criterion of three correct responses to the rule before three incorrect responses, instruction is terminated; the student needs help from the teacher.

<u>Correction</u>

The "Correction" section begins with instruction on editing--why we edit (because we make mistakes when writing) and how we edit (with specific text-editor procedures on the computer).

Then a sentence containing an example of a rule under study is presented. The student is asked whether the sentence is correct, and types "yes" or "no." If the sentence is correct and the student types "yes," another item is presented. If the sentence is correct and the student types "no," the computer tells the student that the sentence is correct and displays the rule being applied in the sentence; then another item is presented.

If the sentence is incorrect and the student types "yes," the computer demonstrates how the sentence should be written and displays the rule; then another item is presented. If the sentence is incorrect and the student types "no," the student is asked to correct the sentence. If appropriately corrected, another item is presented. If not appropriately corrected, the correction is demonstrated by the computer and the appropriate rule displayed; then another item is presented.

Student performance is tallied for "Success" or "Failure" on each rule:

<u>Success</u>: Three "yes" responses to correct sentences <u>and</u> three "no" responses to incorrect sentences, followed by three corrections of these sentences. (Once a student has had "Success" with either correct sentences or incorrect sentences [including appropriate corrections] for a rule, no more items of that type [i.e., correct/incorrect] are presented for that rule.)

<u>Failure</u>: Three inappropriate responses to items (either "no" for correct sentences or "yes" for incorrect sentences) <u>or</u> three inappropriate corrections of incorrect sentences.

When a student has "Success" on all the rules being studied, she/he moves to the "Dictation" section. If the student has "Failure" on one or more rules, additional practice is provided for each rule with "Failure."

After another example of the rule is presented, additional practice consists of more sentences to process as described above (i.e., to determine whether the sentences are correct, and, if incorrect, to correct them). When the student has "Success" on all the rules receiving additional practice, she/he moves to the "Dictation" section. If the student again has "Failure," instruction is terminated; the student needs help from the teacher.

Dictation

For the "Dictation" section of the program, the computer presents sentences orally, and the student types in the sentence.* If the sentence is typed correctly, another item is presented. If the sentence is not typed correctly, the computer checks for errors and prompts the student to correct the sentence. The specific checking routine would depend on the type of rule being practiced; an example of a checking routine for capitalization is as follows:

1. Check for the expected capital letter and prompt student to add a capital letter. (This checking continues for all capital letters needed in the sentence.)
2. Check for a period at the end of the sentence and prompt student to add a period.
3. Check for the correct number of words, prompt student to add or delete words, and repeat sentence orally.
4. Check for individual word matches (misspellings), prompt student to correct word, and repeat sentence orally.

If, after such a checking routine, the sentence is still not correct, the computer displays the correct sentence and then presents another item. Additional items are presented until the student has responded, for each

*Oral presentation of sentences by computer presents minor problems. While tape-recorders provide good aural quality, they do not allow easy random access to the sentences used. Voice synthesizers permit such random access, but the current aural quality of synthesizers apparently leaves something to be desired; however, the quality of voice synthesizers is improving.

rule, with either three sentences with correct rule use or three sentences with incorrect rule use. (Once the student has responded with either three correct or three incorrect rule uses, no additional items for that rule are presented. Note: Although non-rule errors are processed, they are not counted as part of this correctness criterion.) When a student has correctly completed three sentences for all of the rules being studied, instruction for those rules is ended. (And the student studies more rules in the program as needed.) If a student does not correctly complete three sentences for one or more rules, additional practice is provided for such rules.

After another example of the rule is presented, additional practice consists of more dictated sentences. When the student reaches criterion on all of the rules receiving additional practice, instruction is ended for the current set of rules (and the student studies more rules as needed). If the student does not meet criterion during additional practice, instruction is terminated; the student needs help from the teacher.

CONCLUSION

This paper has outlined how instruction in editing might proceed, using a computer to interact with students. Although specific content has not been provided, specifying mechanics content is a relatively straightforward--if time-consuming--task. In addition, computer programming, while also time-consuming, is relatively straightforward, making use of the same or similar routines for a variety of content.

Given sufficient user interest and sufficient resources, such computer-based instruction in editing could be a reality.

SAMPLE SPECIFICATIONS FOR COMPUTER-BASED PRACTICE IN EDITING

Bruce Cronnell

On the following pages are sample specifications for computer-based practice in editing. These specifications elaborate on the general specifications described in WP 2-82/04. In order to illustrate more clearly what such computer-based practice might look like, examples are provided using three (low-level) capitalization rules. (Thus this sample is probably most appropriate for students at the middle- and upper-elementary levels.) In a complete program, content would include more capitalization rules, plus rules for punctuation, spelling, and usage.

These sample specifications do not include the pretest component that is used to determine which rules a student needs to study. Consequently, the sample is designed to operate as if the student were studying only the three rules used for illustration.

Note: The items for this instruction were written by Larry Gentry.

EDITING - CAPITALIZATION

"TITLE"

1. | SWRL PRESENTS |

Display for 5 seconds.*

2. | capitalization |

Graphic: Letters in "capitalization" change, one by one, to capital letters.

Display for one second after end of animation.

If responses are found on student disk:

Go to "REVIEW."

If responses are not found on student disk:

Go to "INTRODUCTION."

*Times are approximate. They can be changed for ease of programming.

"INTRODUCTION"

Frame	Screen text	Notes
1.	Please type your first and last names. Be sure you type correctly. Then push the RETURN key.	Names are filed so computer can call up student's subsequent instruction by name.
2. (SCROLL)	Hi, *<first name>*. Whenever you see *<first name>*? on the screen, you need to type an answer. After you answer, always push the RETURN key. Let's practice. What grade are you in? *<first name>*?	*<first name>*: first name filed in frame 1. The grade number is filed.
3. (SCROLL)	Good, *<first name>*. Before we begin, you need to know how to escape! If you want to stop at any time during the lesson, just type BYE when you see a question mark. Push RETURN.	

NOTE: The word "SCROLL" at the side of the frame indicates that the text is to scroll on the screen, with pauses between paragraphs.

"INTRODUCTION" (continued)

4. We are going to begin some lessons on CAPITALIZATION.

Do you want to review some rules for using capital letters?

Type yes or no.

<*first name*>?

SCROLL

Comment: Student could type only "Y" or "N," but this would make the directions more complex. Computer needs to check only for "Y" ("y") or "N" ("n").

YES: Go to "RULES."

NO: Go to "CHOICE."

"REVIEW"

1. | Please type your first and last names. Then push the RETURN key. |
|---|

If no match with names on file:

a. | I do not know <*first & last name*>. Please type the name you used last time. |
|---|

If still no match:

b. | I still do not find the name <*first & last name*>. Please get your teacher to help me. |
|---|

The computer stops, leaving the display on the screen. Teacher types in the correct name or terminates by removing the disc.

2. SCROLL

> Hello, <*first name*>.
>
> Remember:
>
> - When you see <*first name*>? on the screen, you need to type an answer.
> - After you type your answer, always push RETURN.
> - If you want to stop before the lesson is over, just type BYE when you see a question mark.
>
> Now push RETURN.

"REVIEW" (continued)

SCROLL

3. We are going to do some more work on CAPITALIZATION.

   Do you want to review some rules for using capital letters?

   Type yes or no.

   <*first name*>?

YES: Go to "RULES."

NO: Go to where student left off in last session.

"RULES"

SCROLL

Here are some rules for using CAPITAL LETTERS: 1. The first word in a sentence is always capitalized. My friends are all here. 2. The word "I" is always capitalized. This is where I live. 3. Names of people are capitalized. That nurse is Pat Brown. Push RETURN.

Graphic: Underlined letters are graphically highlighted. (Underlining does not necessarily appear on screen.)

When "RULES" follows "INTRODUCTION":

Go to "CHOICE."

When "RULES" follows "REVIEW":

Go to where student left off in last session.

"CHOICE"

1.

SCROLL

Which sentence is correctly capitalized?

<item>

Type a or b.

<first name>?

Item is displayed; see ITEMS FOR "CHOICE."

2. Correct answer:

Right! Let's try another one.

Display for 5 seconds.

Go to frame 1 (but see constraints on next page).

3. Wrong answer:

SCROLL

Let's review this rule, *<first name>*.

<rule>

Appropriate rule is displayed; see ITEMS FOR "CHOICE."

Look at this sentence:

<sentence>

<comment>

Incorrect sentence from frame 1 displayed. Incorrect lower-case letter is highlighted and changes to capital.

Appropriate comment is displayed; see ITEMS FOR "CHOICE."

Now let's try again.

Push RETURN.

Go to frame 1 (but see constraints on next page).

"CHOICE" (continued)

Student cycles back to frame 1 until:

1. First time (items 1-5):

   Student has responded, for each rule, with either three correct answers or three incorrect answers. (Once the student has responded either correctly to three items for a rule or incorrectly to three items for a rule, no additional items for that rule are presented.)

   a. Student has responded correctly to three items for each of the three rules:

      Go to "EDITING."

   b. Student has responded incorrectly to three items for one or more rules:

      Go to "RULES-SINGLE" for each rule with three incorrect items.

2. Second time (items 6-10):

   a. Student has responded correctly to three items:

      Go to "RULES-SINGLE" for remaining rule(s) missed the first time on "CHOICE."

      If this is the last (or only) rule missed the first time on "CHOICE," go to "EDITING."

   b. Student has responded incorrectly to three items:

      Go to "TERMINATION ROUTINES," frame 7.

ITEMS FOR "CHOICE"

Items are in three categories (A, B, C), based on the rules that they provide practice on. The first time that the student cycles through "CHOICE," categories are presented randomly* in frame 1; only items 1-5 within each category are presented in the order given.

The second time that the student cycles through "CHOICE" (i.e., after going through "RULES-SINGLE"), only the appropriate category for the rule is presented, using items 6-10 in the order given.

Correct answers are asterisked. (Asterisk not to be displayed.)

For frame 3 of "CHOICE," the *<rule>* is that given at the beginning of each category. The *<sentence>* is the incorrect (unasterisked) answer. The underlined letter is graphically highlighted and changes to a capital. The *<comment>* is that given at the end of the category listing.

A. *<rule>* The first word in a sentence is always capitalized.

1. *a. This is my friend's car.
   b. <u>t</u>his is my friend's car.

2. a. <u>y</u>our car is dirty.
   *b. Your car is dirty.

3. a. <u>w</u>here is the fire truck going?
   *b. Where is the fire truck going?

4. a. <u>i</u>t isn't safe to drive fast.
   *b. It isn't safe to drive fast.

5. *a. Many people drive to work.
   b. <u>m</u>any people drive to work.

---

6. *a. People live in many kinds of houses.
   b. <u>p</u>eople live in many kinds of houses.

7. a. <u>s</u>ome houses are made of stone.
   *b. Some houses are made of stone.

8. *a. Other houses are made of brick.
   b. <u>o</u>ther houses are made of brick.

9. a. <u>h</u>ouses are also made of wood.
   *b. Houses are also made of wood.

10. *a. What kind of house do you live in?
    b. <u>w</u>hat kind of house do you live in?

*The same random presentation may be given to each student. However, items are not randomized here because the categories would be harder to recognize.

ITEMS FOR "CHOICE" (continued)

<comment> Since "*<word>*" is the first word in the sentence, it must be capitalized.

[The *<word>* is the word from the sentence for which the first letter is underlined (e.g., "This" in #1); the underlining, of course, does not appear on the screen, but a capital letter is used.]

B. *<rule>* The word "I" is always capitalized.

1. *a. Soon I will learn how to drive.
   b. Soon i will learn how to drive.

2. a. My friend and i rode in the truck.
   *b. My friend and I rode in the truck.

3. *a. When can I ride in your car?
   b. When can i ride in your car?

4. a. That is where i saw the truck.
   *b. That is where I saw the truck.

5. *a. Should I get a big car or a small car?
   b. Should i get a big car or a small car?

---

6. a. During the week, i go to school.
   *b. During the week, I go to school.

7. a. First i get out of bed.
   *b. First I get out of bed.

8. *a. Next I get dressed.
   b. Next i get dressed.

9. *a. Then I eat my breakfast.
   b. Then i eat my breakfast.

10. a. After that i am ready for school.
   *b. After that I am ready for school.

<comment> The word "I" must be capitalized.

ITEMS FOR "CHOICE" (continued)

C. *<rule>* Names of people are capitalized.

1. *a. That toy truck belongs to Pat.
   b. That toy truck belongs to pat.

2. *a. I don't think Henry Appleton can drive.
   b. I don't think henry Appleton can drive.

3. a. The driver is Nancy smith.
   *b. The driver is Nancy Smith.

4. *a. Can Larry Peters ride with you?
   b. Can Larry peters ride with you?

5. a. Does lucy Brown know how to drive?
   *b. Does Lucy Brown know how to drive?

---

6. a. Paul always tries to catch marcia.
   *b. Paul always tries to catch Marcia.

7. a. But she runs after ruth Waters.
   *b. But she runs after Ruth Waters.

8. a. Sometimes Joe black hides when we play.
   *b. Sometimes Joe Black hides when we play.

9. *a. We usually can't catch Sandy Arthur.
   b. We usually can't catch Sandy arthur.

10. *a. My friend Jim is often "it."
    b. My friend jim is often "it."

<comment> Since "*<word>*" is a name, it must be capitalized.

[The *<word>* is the word from the sentence for which the first letter is underlined above (e.g., "Pat" in #1); the underlining, of course, does not appear on the screen, but a capital letter is used.]

"RULES-SINGLE"

SCROLL

Study this rule for using CAPITAL LETTERS: *<rule>* *<sentence>* Push RETURN.

Return to previous section ("CHOICE," "CORRECT," "DICTATION"), using only items for appropriate rule.

<rule> for rule with three incorrect items in "CHOICE" or "DICTATION" or with Failure in "CORRECT":

For A: The first word in a sentence is always capitalized.

For B: The word "I" is always capitalized.

For C: Names of people are capitalized.

<sentence> for appropriate *<rule>*:*

After "CHOICE":

A: Many people live in houses.

B: This is what I do in the morning.

C: My friend Paul Johnson likes to play tag.

After "CORRECT":

A: We had very little snow last winter.

B: What should I have for breakfast?

C: Does Ana have new shoes?

After "DICTATION":

A: There are many kinds of fish.

B: There are many books I can read.

C: Can Nancy Jones climb a tree?

*Underlined letter is graphically highlighted. (Underlining does not necessarily appear on screen.)

"EDITING"*

1. SCROLL

Sometimes when we are writing, we make mistakes.

Sometimes we forget to use a capital letter where we need one.

Look at this sentence.

Where is alice?

Push RETURN.

Sentence remains on screen.

2. SCROLL

This sentence needs a capital letter.

Where is alice?

The name Alice should begin with a capital letter.

Push RETURN.

The a in alice is graphically highlighted.

3. SCROLL

First we move the cursor to the letter that needs to be changed.

Where is alice?

Push RETURN.

Cursor moves under a in alice.

4. SCROLL

Then we type a capital letter.

Where is Alice?

Push RETURN.

The a changes to A and is graphically highlighted.

5. SCROLL

Now the sentence is correct.

Where is Alice?

You can correct sentences the same way.

Push RETURN.

Go to "CORRECT."

*Exact instruction in frame 3 and 4 depends on nature of actual computer editing procedure (and on student experience with computer).

"CORRECT"

1.

SCROLL
Is this sentence correctly capitalized? *<item>* Type yes or no. *<first name>*?

Item is displayed; see ITEMS FOR "CORRECT."

When sentence is correctly capitalized (i.e., no asterisk in ITEMS FOR "CORRECT"):

YES: Go to frame 2.

NO: Go to frame 3.

When sentence is incorrectly capitalized (i.e., asterisk in ITEMS FOR "CORRECT"):

YES: Go to frame 5.

NO: Go to frame 4.

2.

Right! Let's try another one.

Display for 5 seconds.

Go to frame 1 (but see constraints on page 16).

3.

SCROLL
Let's look again, *<first name>*. This sentence is correctly capitalized. *<item>* *<rule>* Now let's try again. Push RETURN.

Same item is displayed; appropriate letter is graphically highlighted.

Appropriate rule is displayed; see ITEMS FOR "CORRECT."

Go to frame 1 (but see constraints on page 16).

"CORRECT" (continued)

4.

SCROLL

You're right, <*first name*>! This sentence is not capitalized correctly. <*item*> Please correct the sentence, and then push RETURN.

Same item displayed.

Correction OK: Go to frame 2.

Correction not OK: Go to frame 5.

5.

SCROLL

Let's look again, <*first name*>. The sentence should be capitalized this way: <*corrected item*> <*rule*> Now let's try again. Push RETURN.

Correct form of item is displayed; appropriate letter is graphically highlighted.

Appropriate rule if displayed; see ITEMS FOR "CHOICE."

Go to Frame 1 (but see constraints on page 16).

"CORRECT" (continued)

Student accuracy in frames 1 and 4 is tallied for Success or Failure on each rule.

<u>Success</u>: 3 YES responses to correct sentences in frame 1, <u>and</u> 3 NO responses to incorrect sentences in frame 1, followed by 3 OK corrections in frame 4

Once student has had success with either correct sentences or with incorrect sentences (including OK corrections) for a rule, no more items of that type (i.e., correct/incorrect) are presented for that rule.

<u>Failure</u>: 3 wrong responses in frame 1 (i.e., 3 responses that are NO to correct sentences and/or YES to incorrect sentences) <u>or</u> 3 not-OK corrections in frame 4

Student cycles back to frame 1 until:

1. First time (items 1-10), after student has achieved either Success or Failure on each of the rules:

   a. Student has achieved Success on all three rules:

      Go to "DICTATION."

   b. Student has achieved Failure one or more rules:

      Go to "RULES-SINGLE" for each rule that student has failed.

2. Second time (items 11-20)

   a. Student has achieved Success on the rule:

      Go to "RULES-SINGLE" for remaining rule(s) missed the first time on "CORRECT."

      If this is the last (or only) failed rule the first time on "CORRECT," go to "DICTATION."

   b. Student has achieved Failure on the rule:

      Go to "TERMINATION ROUTINES," frame 7.

ITEMS FOR "CORRECT"

Items are in three categories (A, B, C), based on the rules that they provide practice on. The first time that the student cycles through "CORRECT," categories are presented randomly* in frame 1; only items 1-10 within each category are presented in the order given.

The second time that the student cycles through "CORRECT" (i.e., after going through "RULES-SINGLE"), only the appropriate category for the rule is presented, using items 11-20 in the order given.

Incorrectly capitalized items are asterisked. (Asterisk not to be displayed.)

For frames 3 and 5, the <*rule*> is the one given at the beginning of the category for the item being discussed. For frame 5, the <*corrected item*> is the item with the underlined lower-case letter changed to an upper-case letter. In frames 3 and 5, the underlined letter, which is always a capital when pointed at, is graphically highlighted.

*The same random presentation may be given to each student. However, items are not randomized here because the categories would be harder to recognize.

ITEMS FOR "CORRECT" (continued)

A. <*rule*> The first word in a sentence is always capitalized.

1. This is my favorite beach.
2. *did you find any shells?
3. Lifeguards must be good swimmers.
4. *where do you work?
5. Some jobs are a lot of fun.
6. Most people like dogs.
7. *can your dog do any tricks?
8. *her dog likes to chase cats.
9. That dog is afraid of cats.
10. *what is your dog's name?

---

11. *some states had a lot of snow.
12. Our relatives are used to cold weather.
13. It often snows in the mountains.
14. *after it snows, we can go skiing.
15. *in the spring, the snow will melt.
16. The trees will be green again.
17. *that's my favorite time of year.
18. Many people prefer warm weather.
19. When it is warm, we can go swimming.
20. *swimming is a lot of fun.

ITEMS FOR "CORRECT" (continued)

A. <*rule*> The word "I" is always capitalized.

1. *My sister and i went to the beach.
2. Last summer, I learned how to swim.
3. *Do you think i can swim that far?
4. *After school, i have a job.
5. My boss said that I can work tomorrow.
6. *My dog wags its tail when i come home.
7. Should I take the dogs for a walk?
8. Tomorrow I will give them a bath.
9. *They were just puppies when i got them.
10. My friend and I built a doghouse.

---

11. Yesterday I didn't eat lunch.
12. Sometimes I eat in a restaurant.
13. *Tomorrow i will eat in the cafeteria.
14. My father and I went to the market.
15. *The banana was ripe when i bought it.
16. *Where will i find the bread?
17. *Can i invite a friend to dinner?
18. My brother and I baked a cake.
19. *The dentist said i can't eat sweets.
20. If you bring fruit, I will bring salad.

ITEMS FOR "CORRECT" (continued)

C. <*rule*> Names of people are capitalized.

1. *Who stepped on mary's sand castle?
2. I met John White at the beach.
3. My friend Sue doesn't like cold water.
4. *I work for gloria Rose.
5. *The best worker is Jack hill.
6. Does Jeff Green have any pets?
7. Have you seen Emily's dog?
8. *The small dog belongs to Robin hill.
9. *I gave a puppy to ricardo.
10. Does Pat's dog have a name?

---

11. You can borrow a shirt from Joe.
12. *Those gloves belong to Bill baker.
13. I think Judy's new dress is pretty.
14. *These pants are too big for eddie.
15. *Where does donna buy her clothes?
16. We bought a belt for Ms. Gray.
17. *Did steve Jones wear a suit?
18. Where does Elena Garcia buy her clothes?
19. That tie won't match Dan's shirt.
20. *In the winter, betsy wears a heavy coat.

"DICTATION"

1. SCROLL

> Now you are going to hear some sentences. You will hear each sentence two times.
>
> Listen to the sentence and type it out correctly. Be sure to use capital letters correctly.
>
> Push RETURN.

2.

> Type the sentence.
>
> <*first name*>?
>
> Push RETURN when you are done.

Synthesizer presents sentence (see ITEMS FOR "DICTATION"), pauses 15 seconds, and repeats sentence.

3. Sentence is correct:

> That's right, <*first name*>. Let's try another one.

Display for 5 seconds.

Go to frame 2 (but see constraints on next page).

4. Sentence is incorrect:

Go to "ITEM-PROCESSING PROCEDURE."

"DICTATION" (continued)

Student cycles back to frame 2 until:

1. First time (items 1-5):

   Student has responded, for each rule, with either three correctly capitalized sentences or three incorrectly capitalized sentences. (Once the student has responded with either three correct capitalizations or three incorrect capitalizations for a rule, no additional items for that rule are presented. Note: Although other errors are processed, they are not counted for purposes of cycling student through "DICTATION."

   a. Student has correctly capitalized three sentences for each of the three rules:

      Go to "TERMINATION ROUTINES," frame 2.

   b. Student has incorrectly capitalized three sentences for one or more rules:

      Go to "RULES-SINGLE" for each rule with three incorrect capitalizations.

2. Second time (items 6-10):

   a. Student has correctly capitalized three items:

      Go to "RULES-SINGLE" for remaining rule(s) missed the first time on "DICTATION."

      If this is the last (or only) rule missed the first time on "DICTATION," go to "TERMINATION ROUTINES," frame 2.

   b. Student has incorrectly capitalized three items:

      Go to "TERMINATION ROUTINES," frame 7.

ITEMS FOR "DICTATION"

Items are in three categories (A, B, C), based on the rules that they provide practice on. The first time that the student cycles through "DICTATION," categories are presented randomly* in frame 2; only items 1-5 within each category are presented in the order given.

The second time that the student cycles through "DICTATION" (i.e., after going through "RULES-SINGLE"), only the appropriate category for the rule is presented, using items 6-10 in the order given.

A. (first word of sentence)

1. This is my house.
2. Many people live in a big city.
3. Games are a lot of fun
4. She is happy now.
5. Apples are good.

---

6. Some fish live in lakes.
7. Other fish live in rivers.
8. Many fish live in the ocean.
9. A whale is not a fish.
10. Whales are big animals.

*The same random presentation may be given to each student. However, items are not randomized here because the categories would be hard to recognize.

ITEMS FOR "DICTATION" (continued)

B. ("I")

1. Here is where I live.
2. Sometimes I like to play outdoors.
3. My mother and I play games.
4. Last night I saw her.
5. This is why I like apples.

---

6. Sometimes I read funny books.
7. These are books that I like.
8. Sometimes I read sad books.
9. Then I cry.
10. You and I can read the same book.

C. (names)

1. My friend Ted lives there.
2. We can play with Jane.
3. That is Sam Brown.
4. Then Bill ate an apple.
5. They gave an apple to Mary.

---

6. Both Jim and Pat play in trees.
7. Once Bob Green made a house in a tree.
8. Then Pam went up in the tree.
9. We could see Jill in the tree.
10. Last year Fred fell from a tree.

ITEM-PROCESSING PROCEDURES

The following steps are completed in sequence; if a condition does not apply, it is skipped. Throughout these proecedures, the student sentence (from "DICTATION," frame 2) remains at the top of the screen; the cursor is at the beginning of the sentence to permit student changes. If, after any change, the student has a correct sentence, go to "DICTATION," frame 3.

1. The capital letter expected for the sentence is not present:*

a. SCROLL

You need to add a capital letter.
Please correct. Then push RETURN.
<*first name*> ?

For item category C only: A second capital letter expected for the sentence is not present:*

b. SCROLL

You need to add another capital letter.
Please correct. Then push RETURN.
<*first name*>?

2. For categories B and C: First word in sentence is not capitalized:

Frame 1 has not be used:

Display frame 1a (above).

Frame 1 has been used:

Display frame 1b (above).

3. No period at end of sentence:

SCROLL

You forgot to put a period at the end of your sentence.
Please add a period. Then push RETURN.
<*first name*>?

*For categories B and C, the first word in the sentence is ignored here.

ITEM-PROCESSING ROUTINES (continued)

4. The number of words is incorrect:

a. The number of words is less than the correct number.

SCROLL	Have you left out any spaces or words? Listen again and fix your sentence. Then push RETURN. *<first name>?*	Wait 10 seconds; then repeat sentence (on synthesizer).

b. The number of words is greater than the correct number:

SCROLL	Have you put in any extra spaces or extra words? Listen again and fix your sentence. Then push RETURN. *<first name>?*	Wait 10 seconds; then repeat sentence (on synthesizer).

5. A word in the sentence does not match the stimulus:

SCROLL	The underlined word is not right. Listen again and fix your sentence. Then push RETURN. *<first name>?*	Mismatched word is underlined (until student makes change). Wait 10 seconds; then repeat sentence (on synthesizer).

Repeat this frame for all mismatched words.

6. The sentence still is not correct:

Repeat steps 1-5.

Sentence still is not correct after repeating steps 1-5:

This is what the sentence should be: *<correct sentence>* Let's try another one.	Display for 20 seconds.

Go to "DICTATION," frame 2 (but see constraints on page 22).

7. After 3 sentences for which a second pass does not result in correct sentence:

Go to "TERMINATION ROUTINES," frame 8.

TERMINATION ROUTINES

Whenever student is terminated--for day or at end of module--all student work is filed (see "FILES AND REPORTS").

1. Student has completed "CHOICE" or "CORRECT" or first time through "DICTATION":

> Good work, <*first name*>. Let's stop for today. We will do some more capitalization later.
>
> Goodbye, <*first name*>.

Display for 10 seconds.

Termination description: "Session ended at conclusion of lesson section."

2. Student has successfully completed "DICTATION":

> DIPLOMA
>
> <*first & last name*> has completed the editing lessons for capitalization.
>
> Congratulations, <*first name*>. You have finished all the lessons.

Graphic: Unscrolled "diploma."

Display for 15 seconds.

Termination description: "Session ended at conclusion of program. STUDENT HAS SUCCESSFULY COMPLETED 'CAPITALIZATION.'"

3. Student types BYE at any time when the computer is waiting for input:

> Goodbye, <*first name*>. We will do some more capitalization later.

Termination description: "Session ended at student's request."

4. Student does not respond within 15 seconds to command/question requiring a typed response:

TERMINATION ROUTINES (continued)

a. I am waiting for your answer.

Display at bottom of screen. Sound is made when sentence is first displayed.

Student still does not respond within 30 more seconds.

b. I'm sorry. Goodbye.

Termination description: "Session ended because student did not respond."

5. Student does not push RETURN within 30 seconds when it must be pushed to continue:

I'm waiting for you to push RETURN.

Display at bottom of screen. Sound is made when sentence is first displayed.

Student still does not respond within 30 more seconds:

Go to next instructional frame (i.e., act as if student had pushed RETURN).

However, if this happens twice (i.e., if student twice goes through two 30-second periods without pushing RETURN):

Display frame 4b (above).

Termination description: "Session ended because student did not push RETURN in order to continue."

6. Student responds to command/question, but does not push RETURN within 15 seconds after response:

Display frame 5 (above).

Student still does not respond within 15 more seconds:

Go to next instructional frame (i.e., act as it student had pushed RETURN).

TERMINATION ROUTINES (continued)

However, if this happens twice (i.e., if student twice responds and twice goes through two 15-second periods without pushing RETURN):

Display frame 4b (above).

Termination description: "Session ended because student did not push RETURN after making response."

7. Student fails the second time on "CHOICE," "CORRECT," OR "DICTATION":

I think you need some more help before we continue. Goodbye, <*first name*>.

Display for 10 seconds.

Termination description: "Session ended because student was unsuccessful the second time on this section. THIS STUDENT PROBABLY NEEDS SPECIAL HELP WITH CAPITALIZATION.

8. Student fails to type correct sentences on "DICTATION" (see step 7 of "ITEM-PROCESSING ROUTINES"):

Display frame 7 (above).

Termination description: "Session ended because student was unable to write sentences from dictation."

NOTE: If the student is terminated by frames 4-8, he/she needs special help: for frames 4-6, help with using the computer; for frames 7-8, help with content. After the teacher has provided appropriate assistance, the student may go back to the computer to continue instruction.

FILES AND REPORTS

The computer keeps track of student responses to individual items and records student time for each session. When a session ends, this information is filed for use in the reports illustrated on the following pages.

EDITING - CAPITALIZATION: STUDENT REPORT

Name: *<first & last name>* School: *<school name>*

Teacher: *<teacher name>* Grade: <#>

	RULE		
	First word in sentence	"I"	Names
CHOICE* (<first/second> time)	[first/second time not noted if student did not have to go through a second time; for second time <#> and % are replaced by --- if student did not have to cover that rule]		
Number of items attempted	<#>	<#>	<#>
Number of successful items	<#> <#>%	<#> <#>%	<#> <#>%

*Repeated for CORRECT and DICTATION.

Note: Similar reports can be made for class, grade, school, etc. Such reports are rather straightforward versions of this report.

EDITING - CAPITALIZATION: DAILY CLASS REPORT

Teacher: *<teacher name>* Grade: <#> School: *<school name>* Date: *<dat*

Student: *<first & last name>*

Session: <#> [Session = each time student is on computer] Session length: <#> minutes

Section: <CHOICE/CORRECT/DICTATION> <First/Second> Time

	RULE		
	First word in sentence	"I"	Names
Number of items attempted	<#>	<#>	<#>
Number of successful items	<#> <#>%	<#> <#>%	<#> <#>%

<Termination Description> [see "TERMINATION ROUTINES"]

Note: At any time, teacher can request a Daily Class Report to look at status of each student in the class. Teacher would be able to determine the following:

1. Student progress through program (which section student is on).
2. Student speed (session length and number of items attempted).
3. Student success (percent of items that are successful)
4. Student problems (when student is terminated for not responding or for not pushing RETURN; when student requests to end session).
5. Student completion of program.

Description

SWRL

SOUTHWEST REGIONAL LABORATORY
WORKING PAPER

DATE: September 30, 1982

NO: WP 2-82/03

COMPUTER INSTRUCTION ON GENERATING IDEAS FOR WRITING DESCRIPTION

Ann Humes

ABSTRACT

Appropriate computer instruction for elementary school students to teach generating ideas for writing description is presented. The initial orientation procedures are discussed. Then the mainline instruction and practice are outlined, and the corresponding branching for students' responses is described. Termination procedures are explained and the screen displays for these procedures are exemplified. A brief conclusion discusses the potential of the instruction.

COMPUTER INSTRUCTION ON GENERATING IDEAS FOR WRITING DESCRIPTION

Ann Humes

Although great advances have been made in computer instruction for the content areas of math and science, courseware for teaching composition is generally limited to the component skills of writing, such as spelling and punctuation. Few programs have been designed to involve students in actual composing activities, yet a need exists for computer instruction that teaches the process of composing.

One of the elements of that process entails generating ideas for writing, and computers can help students master this composing activity. This paper presents appropriate computer instruction on generating ideas for a specific kind of discourse--description. Graphics are the stimulus for the discourse, so they are an essential feature of the program. They are specified here for display on a color monitor. However, these graphics would be enhanced by presenting them by videodisc or videotape if the instructional system has such capabilities.

To explain the design of the computer instruction for elementary school students, this paper first discusses the orientation procedures students undertake after they have loaded the program disk and the student data disk. It then outlines the mainline instruction and practice, describing the corresponding branching that occurs in response to students' input. The next section explains the procedures that terminate the program. This section is followed by a brief conclusion.

ORIENTATION PROCEDURES

At the beginning of the program, students proceed through two of three orientation procedures: the log on, introduction, and review. In the log-on procedure, students are presented first with a screen displaying the name of the developer, followed after an eight-second delay with the name of the program. When the student has used the program before, as determined by data found on the data disk, the computer branches to the review procedure. When the student has not used the program before, the computer branches to the introduction.

The first screen of the introduction, which comprises Section A of the first lesson, displays a request for students to type their first and last names; the computer files these names. The second screen explains the signal for input: the student's name followed by a question mark. It also provides brief practice by requesting that students type in their grade level in response to the signal for input. The next screen in the introduction explains how to stop the program when the computer is waiting for input.

In the review procedure, a screen displays a review of procedures for controlling the program--when to respond, when to push return, and how to exit if the student wishes to exit before the program terminates the lesson. The next display reviews the questions students are taught to use when they generate ideas for writing description (see Table 1). After this review, the computer branches students to their appropriate re-entry point in the lesson.

INSTRUCTION AND PRACTICE

After orientation, the student is presented with the first instruction. Instruction is comprised of three lessons. Lessons 1 and 2 have four sections; Lesson 3 has five sections.

Lesson 1

Section A of Lesson 1 consists of the introduction to the program, as described above. In Section B, the first screen displays an animated picture of popcorn popping, and the second displays a closeup of individual kernels of popcorn. One of these two pictures appears on the screen throughout Section B, depending on which picture is more closely relevant to the text on the screen.

The computer then asks each generating question (see Table 1), and the student generates and types in relevant descriptors as answers to the questions. The computer poses one question per frame so that students' responses can be recorded and evaluated.

Table 1

GENERATING QUESTIONS ASKED ABOUT POPCORN

1.	What does it look like?
2.	What does it sound like?
3.	What does it smell like?
4.	What does it feel like?
5.	What does it taste like?

Each generating question is accompanied by graphic cues that students learn to associate with the corresponding question:

look:	eye
sound:	ear
smell:	nose
feel:	hand
taste:	mouth

If students respond, the computer files the response(s), branches to the Say-More Procedure (see Table 2), and subsequently branches back to the mainline instruction after matching for appropriate descriptors programmed into computer memory. If students do not respond or do not push the return key after their input, the computer branches to the Answer Procedure (see Table 3) and subsequently branches according to students' response to that procedure.

Table 2

SAY-MORE PROCEDURE

Try to type more words.
<first name>
When you are finished, please push the RETURN key.

---> Appropriate cue graphic (i.e., eye, ear, nose, hand, mouth).

When RETURN is pushed, the computer matches words from here, as well as words typed immediately before branching to this procedure, for appropriate set of sensory words, adjusting for spelling errors.[a]

If looks, then white, puffy, like snowflakes.
If sounds, then pop, like rain hitting the roof, like fireworks going off.
If smells, then greasy, delicious.
If feels, then feathery, rough, as light as air, warm.
If tastes, then salty, crunchy, buttery.

If one or more of these words is not matched, unmatched items from appropriate computer list above are displayed in the following frame:

Here *<are some other words/is another word>* that tell(s) *<what popcorn looks like/how popcorn sounds/how popcorn smells/how popcorn feels/how popcorn tastes>*:

<words>

Please push RETURN.

When RETURN is pushed, computer categorizes and files student's words with spelling corrected for matches and any prompted words so that they can be printed in separate lists: student's and computer's. Then computer branches back to next frame in Lesson 1: Section B.

[a]Spelling errors include doubled/non-doubled letters; transposed letters; one missing letter; one extra letter; one wrong letter; inappropriately capitalized words.

Table 3

ANSWER PROCEDURE

I am waiting for your answer. *<first name>*? Press RETURN when you are finished.. --->	Appropriate clue graphic. If the student types in a response, the computer branches to the Say-More Procedure. If the student does not type in one or more words within 30 seconds, the computer displays the following: Here are some words that tell *<question to be answered>* *<words>*: Press RETURN. Computer then displays the list of words that answer the question: how popcorn looks: white, puffy, like snowflakes how popcorn sounds: pop, like rain hitting the roof, like fireworks going off how popcorn smells: greasy, delicious how popcorn feels: feathery, rough, as light as air, warm how popcorn tastes: salty, crunchy, buttery The computer files these words as part of the computer's response to the question and then branches back to next frame in Lesson 1: Section B.

The last two screens in this section display the descriptors that students generate as input for the questions asked on previous screens.

In Section C, students are presented with a picture of bacon and the picture cues for the five senses. Students are then asked to type in the questions that they can use to generate ideas for describing bacon. The computer requests one question per frame so that students' responses can be recorded and evaluated. The graphic cue for each generating question disappears when the student types in the corresponding question. After student input, the computer matches for the words look, sound, smell, feel, taste. If there is a match, the computer branches to the Words Procedure (see Table 4) and subsequently branches back to the main-line instruction. If there is no match or no response, the computer branches to the Assistance Procedure (see Table 5), and subsequently branches to Words Procedure. When students have asked all the questions and generated descriptors as answers to those questions, the computer displays all the generated ideas. The corresponding graphic cues are aligned with the ideas.

In Section D, students select the food they wish to describe, popcorn or bacon. Then the computer displays all the descriptors generated to describe that food. Students obtain a printed copy of the display by pushing P. If a student does not select a topic for composing a description, the computer selects one for him/her. Students then compose their descriptions offline, on regular paper.

Lesson 2

In Section A of Lesson 2, the computer explains that all the sensory questions cannot be posed and answered for many objects the writer may

Table 4

WORDS PROCEDURE

A. | Now type all the words you can think of that tell <appropriate question>, <first name>? <clue graphic>

Please push RETURN when you are finished.

--> Computer displays this frame with whichever of the clues and following inserts is appropriate:

LESSON 1:

bacon: what bacon looks like, how bacon sounds when it is cooking, how bacon smells, how bacon feels when you pick it up, how bacon tastes.

LESSON 2:

rose: what a rose looks like, how a rose smells, how a rose feels when you touch it.

trumpet: what a trumpet looks like, what a trumpet sounds like, what a trumpet feels like when you touch it.

orange: what an orange looks like, what an orange smells like, what an orange feels like when you touch it, what an orange tastes like.

LESSON 3:

dupla: about the shape, size, and color of dupla.

If the student types one or more words, the computer displays the following on the screen with words student has typed:

B. | Can you say more, <first name>? (clue graphic)

When you are finished, push RETURN.

When RETURNED is pushed, student's words are filed for later printing. Then the computer branches as follows:

bacon: to frame 2, Section C, Lesson 1, unless this is last sensory match. If last sensory match, computer branches to frame 3.

rose: to frame 6, Section A, Lesson 2, unless this is the last sensory match. If last sensory match, computer branches to next section.

trumpet: to frame 3, Section B Lesson 2, unless this is the last sensory match. If last sensory match, computer branches to next section.

orange: to frame 3, Section C, Lesson 2, unless this is the last sensory match. If last sensory match, computer branches to next section.

dupla: to frame 3, Section B, Lesson 3, unless this is the last sensory match. If last sensory match, computer branches to next section.

If the student does not type one or more words, the computer displays the following [see the list of words below, for list of questions, see A above]:

B1 | Here is a way to tell <appropriate question>: <word>

The computer then files computer's word and branches as described above.

bacon

- looks: striped
- sounds: sizzling
- smells: smoky
- feels: greasy
- tastes: salty

rose

- looks: white
- smells: like perfume
- feels: velvety

trumpet

- looks: shiny
- sounds: brassy
- feels: cold

orange

- looks: dimpled
- smells: tangy
- feels: sticky
- tastes: sweet

dupla [fantasy figure 2]

- shape: like an ice cream cone
- size: large
- color: orange

Table 5

ASSISTANCE PROCEDURE

Here is a question you can ask: <*appropriate question*> [*appropriate sense graphic*] Please push RETURN.	-->	The computer displays one of the questions that has not yet been asked about the topic: For eye graphic: What does it look like? For ear graphic: What does it sound like? For nose graphic: What does it smell like? For hand graphic: What does it feel like? For mouth graphic: What does it taste like? For shape graphic: What shape is it? For size graphic: What size is it? For color graphic: What color is it? The computer then branches to Words Procedure.

wish to describe. To demonstrate the inappropriateness of some questions, a rose is displayed as the object to be described while graphics dramatize the inappropriate nature of some sensory questions (e.g., a person "tasting" a rose).

The computer then elicits input from students on the questions that can be appropriately posed for a rose. The computer evaluates students' responses and branches when the answer is appropriate, as described for Section C in Lesson 1 above. If there is no match or no-response, the computer branches to the Prompting Procedure (see Table 6).

In Section B, students input questions that are appropriate to ask when they are generating content for describing a trumpet. As in Section A of this Lesson, the computer evaluates students' responses and branches appropriately to either the Words Procedure or the Prompting Procedure. The instructional procedures are the same for Section C, which presents an orange as the object to be described.

Table 6

PROMPTING PROCEDURE

Here is a clue to a question you can ask: /picture: eye/ear/nose/hand/mouth/ *<first name>*?	——> Computer displays an appropriate clue graphic. If this is the first question for the topic, the computer displays first graphic in sequence. If this is not the first question, the computer displays the first graphic in sequence for which a question has not been asked: rose: eye, nose, hand trumpet: eye, ear, hand orange: eye, nose, hand, mouth fantasy figure 2: shape, size, color If the student types in a response, the computer matches for any of the appropriate questions not previously matched. If there is a match, the computer branches to the Words Procedure. If there is no match, or if the student does not respond, the computer branches to the Assistance Procedure.

In Section D, a review of the questions to ask when generating ideas for descriptions is displayed, along with the corresponding graphics cues. The students again select a topic (either a rose, a trumpet, or an orange) to describe offline, and the computer provides a hard copy of the descriptors that students have generated.

<u>Lesson 3</u>

In Section A, students are presented with a fantasy figure (see Figure 1), and computer explains that sometimes the only appropriate generating question is "What does it look like?" The computer then presents specific "subquestions" students can ask when they generate details about how something looks:

What size is it?

What shape is it?

What color is it?

Corresponding graphics cues are displayed. Each cue consists of two figures that are identical except for one feature:

Cues	Figures
size	one figure is larger than the other
shape	one figure is thinner than the other
color	the members of the pair differ only in color

The computer then asks each size, shape, color question, and students generate ideas in response to the question. The computer matches students' input for appropriate descriptors in the program's memory. A descriptor is provided if the student does not generate any appropriate responses.

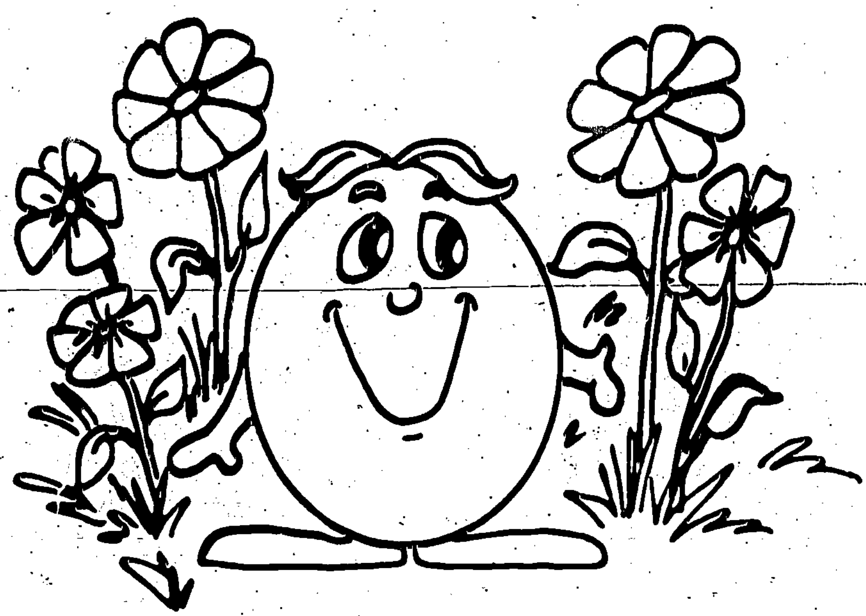


Figure 1: First Fantasy Figure in Instruction

In Section B, the student is presented with two versions of another fantasy figure (see Figure 2). One is a complete figure; the other is a collapsed outline of the same figure. The latter figure becomes the right size, shape, and color as the student inputs the corresponding question. When the student types in an appropriate generating question, the computer matches for the words "size," "shape," and "color." When a match is found, the computer branches to the Words Procedure to elicit appropriate descriptors. When there is either no match or no response, the computer branches to the Assistance Procedure. This process is repeated until the student has typed in all three questions about size, shape, and color.

In Section C, students are taught a further refinement of the size, shape, and color questions. A rabbit is displayed and students learn to ask about the size, shape, and color of its parts (e.g., "What shape are its parts?"). The co ter then has the student generate ideas for describing the size, shape and color of those parts, and responses are evaluated for matches with appropriate descriptors programmed in memory. If the student does not input appropriate descriptors, the computer provides a single descriptor; providing more than one descriptor might encourage students to let the computer do all the generating.

In Section D, students are presented with a spider and asked to identify the parts that can be described in terms of their shapes. The computer then evaluates students' responses. If there is neither a match nor a response, the computer provides the information. Then students are presented with a review screen displaying descriptors of the size, shape, and color of the objects presented in Lesson 3. Students choose their topic for composing offline, as described for Lessons 1 and 2.

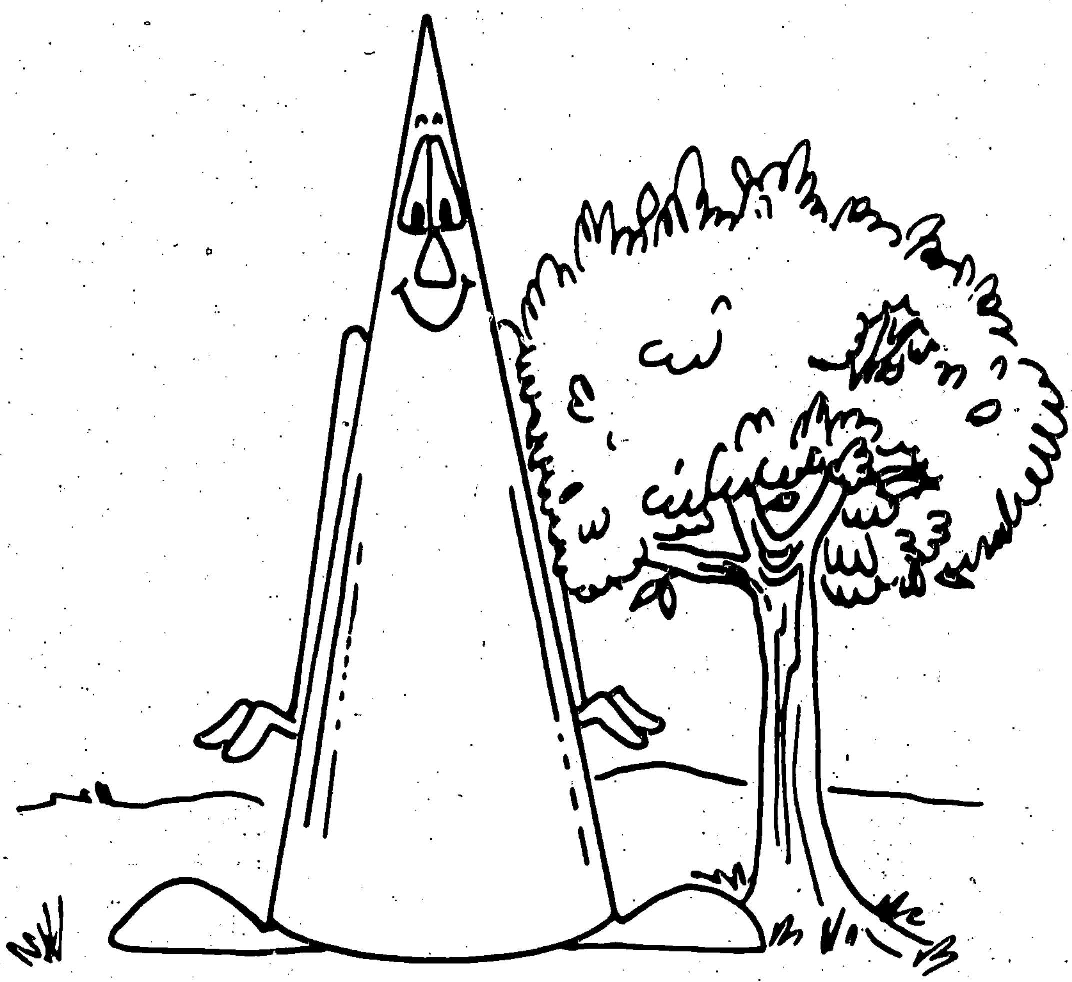


Figure 2: Second Fantasy Figure in Instruction

TERMINATION PROCEDURES

The program can be terminated for several reasons and in several ways. If the student's operating time on the computer is close to 20 minutes, the computer, at the point where it is about to begin a new topic, displays the following screen:

> That is enough for now. We will talk another time
> about getting ideas for writing descriptions.
> Goodbye, <*first name*>.

The computer then files all the student's work during the operating time and logs the student off the computer.

If the student types BYE at any time when the computer is waiting for input, the computer responds with the following screen:

> Goodbye, <*first name*>. We will work later on
> getting ideas for description.

As before, the computer files all the student's work during the operating time and logs the student off the computer.

At any point where the student responds to a command/question but does not push the return key within 30 seconds after the response, the computer displays the following screen:

> If you have finished, please push the RETURN key.
> If you do not, I must say goodbye.

If the student pushes the return key within 15 seconds, the computer branches back to the point where the student was in the lesson before this termination procedure. If the student does not push the return key within 15 seconds, the computer replaces the above frame with the following frame:

I'm sorry. Goodbye, <*first name*>.

The computer then terminates the lesson, filing work student has completed at that time.

CONCLUSION

The paper has described instruction for teaching students how to generate ideas for writing descriptions. It has also presented specifications for branching procedures that provide the potential for a highly interactive program. Furthermore, computer programming for the instruction and branching procedures is feasible, even within current memory limitations, particularly if videotape or videodisc is used to display graphics. Given sufficient user interest and sufficient resources, this instructional design could become a viable and valuable component of a larger instructional package for teaching the composing process.

SAMPLE SPECIFICATIONS FOR COMPUTER INSTRUCTION ON GENERATING IDEAS FOR WRITING DESCRIPTION

Ann Humes

On the following pages are sample specifications for computer instruction on generating ideas for writing descriptions. These elaborate on the general specifications described in WP 2-82/03 and include drafts of special procedures that are called up for branching purposes (pp. 1-10), three lessons on generating ideas (pp. 11-40), and draft graphics for the lessons (pp. 41-47).

Some of the content is subject to change; e.g., better descriptors may replace those used to describe the figures; the current descriptors should be considered placeholders. Also, more appropriate objects may be substituted for those described by the attached drawings.

LOG-ON PROCEDURE

1. SWRL Educational Research and Development

Presents

(Delay 8 seconds before next display.)

2. Getting Ideas for Writing Descriptions

SWRL, Los Alamitos, California

Cartoon picture of a middle-school student from shoulders up with thought balloon above his/her head. Thought balloon contains a light bulb blinking on and off.

If student's responses are found on disk, computer branches to Review Procedures. If no responses are found on the disk, computer goes to Lesson 1: Section A.

REVIEW PROCEDURE

> Please type your first and last names.
> Then push the RETURN key.

Computer uses first and last names to verify that student has his/her own disk. If no match is found, the computer displays the following:

1A

> I do not know <*first and last names*>.
> Please type the name you used last time.

If the student does not change the name or if the response still does not match the name on the disk, the following frame replaces 1A above:

1B

> I still do not find the name <*first and last names*>.
> Please get your teacher to help me.

The computer stops, leaving the display on the screen. Teacher or other adult will terminate by removing the disk or will type in the correct name.

Hello, <*first name*>.

Remember:

- When you see <*first name*>? on the screen I am asking for an answer.
- After you type your answer, always push RETURN.
- If you want to stop before the lesson is over, just type BYE when you see your name and a question mark.

2. Now push RETURN.

Remember that asking these questions will help you get ideas for describing something:

- What does it look like? [eye]
- What does it sound like? [ear]
- What does it smell like? [nose]
- What does it feel like? [hand]
- What does it taste like? [mouth]

3. Now push RETURN.

Computer locates the lesson and section at which the student left off in previous session. It then goes to the first frame in that section.

> I am waiting for your answer.
>
> *<first name>*?
>
> Press RETURN when you are finished.

Appropriate clue graphic.

If the student types in a response, the computer branches to the Say-More Procedure.

If the student does not type in one or more words within 30 seconds, the computer displays the following:

1A

> Here are some words that tell *<question to be answered>*:
>
> *<words>*
>
> Press RETURN.

Computer then displays the list of words that answer the question:

how popcorn looks: white, puffy, like snowflakes

how popcorn sounds: pop, like rain hitting the roof, like fireworks going off

how popcorn smells: greasy, delicious

how popcorn feels: feathery, rough, as light as air, warm

how popcorn tastes: salty, crunchy, buttery

The computer files these words as part of the computer's response to the question and then branches back to next frame in Lesson 1: Section B.

Screen	Computer action
Try to type more words. *<first name>* When you are finished, please push the RETURN key.	Appropriate cue graphic (i.e., eye, ear, nose, hand, mouth).

When RETURN is pushed, the computer matches words from here, as well as words typed immediately before branching to this procedure, for appropriate set of sensory words, adjusting for spelling errors.*

- If looks, then <u>white</u>, <u>puffy</u>, <u>like snowflakes</u>.
- If sounds, then <u>pop</u>, <u>like rain hitting the roof</u>, <u>like fireworks going off</u>.
- If smells, then <u>greasy</u>, <u>delicious</u>.
- If feels, then <u>feathery</u>, <u>rough</u>, <u>as light as air</u>, <u>warm</u>.
- If tastes, then <u>salty</u>, <u>crunchy</u>, <u>buttery</u>.

If one or more of these words is not matched, unmatched items from appropriate computer list above are displayed in the following frame:

> Here *<are some other words/is another word>* that tell(s) *<what popcorn looks like/how popcorn sounds/how popcorn smells/how popcorn feels/how popcorn tastes>*:
>
> *<words>*
>
> 1A Please push RETURN.

When RETURN is pushed, computer categorizes and files student's words with spelling corrected for matches and any prompted words so that they can be printed in separate lists: student's and computer's. Then computer branches back to next frame in Lesson I: Section D.

*Spelling errors include doubled/non-doubled letters; transposed letters; <u>a/an</u>; one missing letter; one extra letter; one wrong letter; inappropriately capitalized/uncapitalized words.

rt A

> Here is a clue to a question you can ask:
>
> /picture: eye/ear/nose/hand/mouth /
>
> *<first name>*?

Computer displays an appropriate clue graphic. If this is the first question for the topic, the computer displays first graphic in sequence. If this is not the first question, the computer displays the first graphic in sequence for which a question has not been asked:

rose: eye, nose, hand
trumpet: eye, ear, hand
orange: eye, nose, hand, mouth

dupla: shape, size, color

If the student types in a response, the computer matches for any of the appropriate questions not previously matched. If there is a match, the computer branches to the Words Procedure.

Part B. If there is no match, or if the student does not respond, the computer displays one of the questions* that has not yet been asked about the topic:

> Here is a question you can ask:
>
> *<appropriate question>* [*appropriate sense graphic*]
>
> Please push RETURN.

The computer then branches to Words Procedure.

*For eye graphic: What does it look like?
For ear graphic: What does it sound like?
For nose graphic: What does it smell like?
For hand graphic: What does it feel like?
For mouth graphic: What does it taste like?
For shape graphic: What shape is it?
For size graphic: What size is it?
For color graphic: What color is it?

A.

> Now type all the words you can think of that tell <*appropriate question*>. <*first name*>? <*clue graphic*>
>
> Please push RETURN when you are finished.

Computer displays this frame with whichever of the clues and following inserts is appropriate:

LESSON 1:

bacon: what bacon looks like, how bacon sounds when it is cooking, how bacon smells, how bacon feels when you pick it up, how bacon tastes.

LESSON 2:

rose: what a rose looks like, how a rose smells, how a rose feels when you touch it.

trumpet: what a trumpet looks like, what a trumpet sounds like, what a trumpet feels like when you touch it.

orange: what an orange looks like, what an orange smells like, what an orange feels like when you touch it, what an orange tastes like.

LESSON 3:

dupla: about the shape, size, and color of dupla.

If the student types one or more words, the computer displays the following on the screen with words student has typed:

B.

> Can you say more, <*first name*>? (clue graphic)
>
> When you are finished, push RETURN.

When RETURNED is pushed, student's words are filed for later printing. Then the computer branches as follows:

bacon: to frame 2, Section C, Lesson 1, unless this is last sensory match. If last sensory match, computer branches to frame 3.

rose: to frame 6, Section A, Lesson 2, unless this is the last sensory match. If last sensory match computer branches to next section.

trumpet: to frame 3, Section B Lesson 2, unless this is the last sensory match. If last sensory match, computer branches to next section.

orange: to frame 3, Section C, Lesson 2, unless this is the last sensory match. If last sensory match, computer branches to next section.

dupla: to frame 3, Section B, Lesson 3, unless this is the last sensory match. If last sensory match, computer branches to next section.

If the student does not type one or more words, the computer displays the following [see below the list of words; for list of questions, see A above]:

B1

Here is a word that tells *<appropriate question>*:
<word>

The computer then files computer's word and branches as described above.

<u>bacon</u>

looks: striped
sounds: sizzling
smells: smoky
feels: greasy
tastes: salty

<u>rose</u>

looks: white
smells: like perfume
feels: velvety

<u>trumpet</u>

looks: shiny
sounds: brassy
feels: cold

<u>orange</u>

looks: dimpled
smells: tangy
feels: sticky
tastes: sweet

<u>dupla</u>

shape: like an ice cream cone
size: large
color: [ ]

TERMINATION PROCEDURES

If the student's operating time on the computer is close to 20 minutes, the computer, at the point where it is about to begin a new topic (i.e., bacon, rose, ______, or ______), will display the following:

1.

That is enough for now. We will talk another time about getting ideas for writing descriptions. Goodbye, <*first name*>.

The computer will file all the student's work during the operating time and log the student off the computer.

If the student types BYE at any time when the computer is waiting for input, the computer will respond with the following:

2.

Goodbye, <*first name*>. We will work later on getting ideas for description.

Then the computer will file the student's work during the operating time and log the student off the computer.

At any point where the student responds to a command/question but does not push RETURN key within 30 seconds after the response, the computer displays the following:

3A

If you have finished, please push the RETURN key. If you do not, I must say goodbye.

If the student pushes the RETURN key within 15 seconds, the computer branches back to the point where the student was in the lesson before this termination procedure.

If the student does not push the RETURN key within 15 seconds, the computer replaces the above frame with the following frame:

3B

I'm sorry. Goodbye, <*first name*>.

The computer then terminates the lesson, filing work student has completed at that time.

LESSON 1: SECTION A (INTRODUCTION)

Screen	Explanation
Please type your first and last names. Be sure you type correctly. Then push the RETURN key.	Name is filed so computer can call up student's subsequent instruction by name.
Hi *<first name>*. Whenever you see *<first name>* ? on the screen, I am asking for an answer. After you answer, always push the RETURN key.	Student's first name is displayed in angled brackets.
Let's practice. What grade are you in? *<first name>* ?	The grade number is filed.
Good *<first name>*. Before we begin, I will tell you how to escape! If you want to stop at any time during the lesson, just type BYE when you see your name and a question mark. I'll get the message. Push RETURN.	The student's first name is displayed on the screen.

LESSON 1: SECTION B (POPCORN)

1.	Now you are going to learn how to get ideas for writing descriptions. Then you will use your ideas to write a description. Please push RETURN.	Repeat thought-balloon graphic used earlier.
2.	Here is a picture of popcorn. <Popcorn Picture A>	Popcorn Picture A: Popcorn popping; popping sounds occur simultaneously with the explosion of kernels on the screen. Picture is labeled "POPCORN." Delay seven seconds before next screen.
3.	Here is another picture of popcorn. <Popcorn Picture B>	Popcorn Picture B: Several individual pieces of popcorn. Picture is labeled "POPCORN." This picture remains on the screen for frame B-4 and for corresponding branching for "look" to Answer Procedure and Say-More Procedure. The label does not stay. Delay seven seconds before next screen.

12

Now let's describe popcorn. What does popcorn look like? Type all the words you can think of that tell what popcorn looks like. Then push the RETURN key.

4. <*first name*> ?

Eye graphic.

If student does not begin typing within 30 seconds, the computer branches to the Answer Procedure.

If the student pushes RETURN but has not typed in any words, the computer branches to the Answer Procedure.

If the student pushes RETURN and if he or she has written one or more words, the computer files the words and then branches to the Say-More Procedure.

What does popcorn sound like? Type all the words you can think of that tell how popcorn sounds when it is popping.

5. <*first name*> ?

Popcorn Picture A. This picture remains on screen during corresponding branching for "sound" to Answer Procedure and Say-More Procedure.

Ear graphic.

If the student does not begin typing within 30 seconds, the computer branches to the Answer Procedure.

If the student pushes RETURN but has not typed in any words, the computer branches to the Answer Procedure.

If the student writes one or more words, the computer branches to the Say-More Procedure.

13

Popcorn Picture B. This picture remains on screen for frames B-6, 7, and 8 and for corresponding branching through Answer Procedure and Say-More Procedure.

6. What does popcorn smell like? Type all the words you can think of that tell how popcorn smells.
<*first name*> ?

Nose graphic.

If the student does not begin typing within 30 seconds, the computer branches to the Answer Procedure.

If the student pushes RETURN but has not typed in any words, the computer branches to the Answer Procedure.

If the student writes one or more words, the computer branches to the Say-More Procedure.

7. What does popcorn feel like when you touch it. Type all the words you can think of that tell how popcorn feels when you pick it up in your hand.
<*first name*> ?

Hand graphic.

If the student does not begin typing within 30 seconds, the computer branches to the Answer Procedure.

If the student pushes RETURN but has not typed in any words, the computer branches to the Answer Procedure.

If the student writes one or more words, the computer branches to the Say-More Procedure.

8. What does popcorn taste like? Type all the words you can think of that tell how popcorn tastes.
<*first name*> ?

Mouth graphic.

If the student does not begin typing within 30 seconds, the computer branches to the Answer Procedure.

If the student pushes RETURN but has not typed in any words, the computer branches to the Answer Procedure.

If the student writes one or more words, the computer branches to the Say-More Procedure.

When we asked questions about what popcorn looks, sounds, smells, feels, and tastes like, we got these words:

<words for looks>

9. Push RETURN.

Popcorn picture B remains on screen for frames 7 and 8.

Eye graphic on same line with words.

<words for sounds>

<words for smells>

<words for feels>

<words for tastes>

10. Push RETURN.

Ear graphic on same line with words

Nose graphic on same line with words

Hand graphic on same line with words

Mouth graphic on same line with words

Screen	Computer action
[Picture: Bacon] Here is a picture of bacon. [Picture clues for all 5 questions.] The other pictures give clues to the questions you can ask. What is one question you can ask to get ideas for writing. *<first name>* ? 1. Push RETURN when you have asked ONE question.	Picture of bacon, in frying pan, is top center of screen. It remains on screen for rest of Section C, including branching. Picture clues for all five questions are across the screen below bacon and the line identifying bacon. When RETURN is pushed, computer matches for sensory words look, sound, smell, feel, taste. If there is a match, corresponding clue leaves the screen. If there is no match, if the student does not begin typing within 30 seconds, or if the student pushes RETURN without typing, computer branches to Part B of Prompting Procedure. If there is a match, computer branches to Words Procedure.
[Picture: Bacon] [Remaining picture clues for senses] What is one other question you can ask? Remember to look at the clues. *<first name>* ? 2. Press RETURN when you have typed another question.	When RETURN is pushed, computer matches for the unused sensory words. If there is a match, the corresponding clue leaves the screen. Clue picture explodes into another part of the bacon. Then the computer branches to Words Procedure. If there is no match, if the student does not begin typing within 30 seconds, or if the student pushes RETURN without typing, computer branches to Part B of Prompting Procedure.

: SECTION C (continued)

let's look at the ideas for writing a
cription of bacon.

we asked questions about how bacon
s, sounds, smells, feels, and tastes,
ot these words:

<words for looks>

RETURN to continue.

Bacon picture remains on screen for frames 3 and 4.

Eye graphic on same line with words.

<words for sounds>

<words for smells>

<words for feels>

<words for tastes>

RETURN.

Clue graphics on same line as corresponding words.

LESSON 1: SECTION D

Now you can write a description of popcorn or bacon. Type the name of the one you want to describe. Your ideas will come on the screen.

1. <*first name*> ?

If the student does not type in one of the topics, the computer displays the following:

Please type in POPCORN or BACON. If you do not type in one of these words, I will choose your topic.

1A. <*first name*> ?

If the student still does not type in one of the topics, the computer displays the following:

You may write on <*random choice of the two topics*>.

[Picture of topic.]

1B. Press RETURN

When student types in popcorn or bacon, or when computer selects popcorn or bacon, the corresponding picture appears on screen and stays or screen through the rest of Section D.

Here are the ideas, <*first name*>:

<*words for each question for the chosen topic*>

2. Please press RETURN.

Corresponding clue graphic appears on line with appropriate words.

Now write your description on paper, using the ideas you have for writing. Push P and all the ideas will be printed on paper.

3. Have fun, <*first name*>.

When P is pushed, computer prints list of words split in two sets. The first set is headed <*first name's*> List and the second is headed Computer's List. Computer then terminates the lesson. [Note: Hard copy of picture will be available for student's reference while he/she composes.]

	Screen		Notes
1.	Sometimes you don't want to or can't ask all five questions about the thing you are describing. Please press RETURN.	→	All clue graphics on screen. Then all but eye graphics leave the screen. Eye graphic leaves screen when student pushes RETURN.
2.	Here is a rose. <*picture*>	→	Picture of a rose with label "ROSE." It is large and fills the center of the screen. Delay seven seconds before next screen.
3.	You would not ask what this rose tastes like unless you are very strange and eat roses. Press RETURN.	→	Picture of head in Log-on Procedure, about to eat a rose.
4.	You would not ask how the rose sounds.	→	Picture of same head, hand held to ear next to rose, musical notes on screen. Delay seven seconds before next frame.
5.	What is one question you would ask? <*first name*>? Press RETURN when you have written ONE question.	→	Picture of rose in frame 2 above, but reduced in size and no label. The computer matches the student's response for key words <u>look</u>, <u>smell</u>, <u>feel</u>. If there is a match, the computer branches to Words Procedure. If there is no match, the computer goes to the Prompting Procedure. If the student does not begin to type a response within 30 seconds or if the student pushes RETURN without typing, the computer branches to the Prompting Procedure.

20

	What is another question you would ask about this rose to get ideas for describing it? <*first name*>?	The computer matches the student's response for key words not already matched. If there is a match, the computer branches to Words Procedure. If there is no match or if the student pushes RETURN without typing, the computer branches to Prompting Procedure. If the student does not begin to type a response within 30 seconds, the computer branches to the Prompting Procedure.
6.	Press RETURN when you have written another question.	

21

Screen	Notes
Here is another picture to ask questions about, but you wouldn't ask how this trumpet smells. 1. Press RETURN.	Picture of trumpet, labeled "TRUMPET" remains on screen for all of section B, including branches. Label leaves screen after this frame.
What is one question you would ask about this trumpet to get ideas for describing it? <*first name*>? 2. When you have typed ONE question, press RETURN.	The computer matches the student's response for key words <u>look</u>, <u>sound</u>, <u>feel</u>. If there is a match, the computer branches to Words Procedure. If there is no match or if the student presses RETURN without typing a response, the computer goes to the Prompting Procedure. If the student does not begin to type a response within 30 seconds, the computer goes to the Prompting Procedure.
What is another question you would ask about this trumpet to get ideas for describing it? <*first name*>? 3. Push RETURN when you have typed another question.	The computer matches the student's response for key words not already matched. If there is a match, the computer branches to Words Procedure. If there is no match or if the student presses RETURN without typing a response, the computer branches to Prompting Procedure. If the student does not begin to type in a response within 30 seconds, the computer branches to the Prompting Procedure.

1.	Here is one more picture to ask questions about.	→ Picture of an orange, labeled "ORANGE," remains on screen for rest of Section C, including branches. Delay ten seconds before next screen.
2.	What is one question you would ask about this orange to get ideas for describing it? *<first name>*? Please push RETURN when you have typed ONE question.	→ The computer matches the student's response for key words <u>looks</u>, <u>smells</u>, <u>feels</u>, <u>tastes</u>. If there is a match, the computer branches to Words Procedure. If there is no match or if the student presses RETURN without typing, the computer goes to the Prompting Procedure. If the student does not begin to type a response within 30 seconds, the computer goes to the Prompting Procedure.
3.	What is another question you would ask about this orange to get ideas for describing it? *<first name>*? Please push RETURN when you have typed another question.	→ The computer matches the student's response for key words not already matched. If there is a match, the computer branches to Words Procedure. If there is no match or if the student presses RETURN without typing, the computer branches to the Prompting Procedure. If the student does not begin to type in a response within 30 seconds, the computer branches to the Prompting Procedure.

Now you know how to ask yourself these questions in order to get ideas for writing descriptions: $\boxed{\text{eye}}$ What does it look like? What does it sound like? $\boxed{\text{ear}}$ $\boxed{\text{nose}}$ What does it smell like? What does it feel like? $\boxed{\text{hand}}$ $\boxed{\text{mouth}}$ What does it taste like? 1. Press RETURN.	→ Clue graphics on same lines.

Now you can write a description of a rose, a trumpet, or an orange. Type the name of the one you want to describe. Your ideas will come on the screen. <*first name*>? 2. Press RETURN.	→ If the student does not type in one of the topics, the computer displays the following:

2A	Please type in rose, trumpet, or orange. If you do not type in one of these words, I will choose your topic. Press RETURN when you are finished. <*first name*>?

If the student still does not type in one of the topics, the computer displays the following:

2B	You may write about <*random choice of the three topics*>. Please press RETURN.

When student types in a topic or when computer selects a topic, the corresponding picture appears on the screen and stays on the screen through the rest of Section D.

24

Screen	Notes
Here are the ideas, <*first name*>: <*words for each "question" for the chosen topic*> 3. Please press RETURN.	Corresponding cue graphic (e.g., eye) appears on line with appropriate words.
Now you can write your description on paper. You will see how easy it is to write when you get your ideas first. Push P and all the words will be printed on paper. 4. Good luck, <*first name*>.	When P is pushed, computer prints list of words split in two sets. The first set is headed <*first name's*> List and the second is headed Computer's List. Computer then terminates the lesson.

	Screen		Notes
1.	*<Picture of fantasy figure>*	→	Picture of a fantasy figure standing among flowers. The figure is smaller than the flowers. The word TERBIE is presented below the picture. The picture covers most of the center of the screen.
2.	Sometimes when you describe something, you can ask yourself only one question: What does it look like? Please push RETURN.	→	Eye graphic clue on line with question.
3.	Here is a picture of an imaginary animal called a terbie. When you want to describe what this terbie looks like, you can ask yourself these questions: What shape is it? /clue pic./ What size is it? /clue pic./ What color is it? /clue pic./ Press RETURN.	→	Reduced picture of fantasy creature remains on screen throughout Section A, including branching. Questions appear on screen one at a time with pause before first question and pause before each subsequent question. A bell rings as each question appears on the screen. The clue picture comes on the screen with the corresponding question. Clue picture for shape is two stick figures same color as letters on screen, identical except that one has a round body and one has a thin, stick body. Clue picture for size is two stick figures, with same color as letters on screen, identical except that one is tall and one is short. Clue picture for color is two stick figures, identical except for different color, which is not the color of screen.

4. Let's practice. Write all the words you can think of that answer the question: What shape is the terbie? /clue pic./

Press RETURN.

Clue picture for shape.

When the student pushes RETURN, computer matches for the word "round." If there is a match, the computer displays the following:

4A Good, <*first name*>. You have some good ideas for describing the shape of the terbie.

If there is no match, if the student has not typed any words within 30 seconds, or if the student presses RETURN without typing, the computer displays the following:

4B Here is one way to describe the shape of the terbie: round /clue picture/

Press RETURN.

Computer files student's responses and computer's response (if any) separately for later display and printing for student.

Now answer this question:

What size is the terbie /clue pic./ *<first name>* ?

5. Press RETURN when you are finished.

Picture for size.

When the student pushes the RETURN, computer matches for the words small, little, tiny. If there is a match for one of these words, the computer displays the following:

5A

> You are doing well, *<first name>*.
> You have some good ideas for describing the size of the terbie.
>
> Press RETURN.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student pushes RETURN without typing, the computer displays the following:

5B

> Here is a good way to describe the size of the terbie: small/clue pic./
>
> Press RETURN.

The computer files student's responses and computer's response (if any) separately for later display to the student.

6. Now here is one more question:

What color is the terbie, *<first name>* ?

Press RETURN when you are finished.

Graphic for color.

When the student pushes the RETURN, computer matches for the word ____. If there is a match, the computer displays the following:

6A Great, *<first name>*! Now let's do something different before you write another description.

Press RETURN.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student presses RETURN without typing, the computer displays the following:

6B Here is a good way to describe the color of the terbie: ____.

Now let's do something different before you write another description. Press RETURN.

Computer files responses as described for frames 4A and 4B above.

1. Here is a picture of an imaginary animal called a dupla. The other picture is also supposed to be a dupla, but it doesn't look like one now. It will become a dupla as you tell how to describe the first dupla.

Press RETURN.

Picture of a fantasy creature next to a tree. The picture is labeled "DUPLA." The creature is taller than the tree. The "outline" of a second, without color and with slightly collapsed shape, is also on the screen. It is smaller. Pictures of creature remain on screen throughout Section B and corresponding branching.

2. What is one question you can ask about how the dupla looks?

<first name> ?

Push RETURN when you have typed one question.

Picture clues for the shape, size, and color questions come on the screen.

When RETURN is pushed, computer matches for words shape, size, color. If there is a match, the corresponding clue graphic moves over to the "outlined" creature, flashes on and off and then disappears as the creature changes according to the match:

If the match is for shape, the creature changes shape to match the other, completed creature on the screen.

If the match is for size, the creature grows to match the size of the other, completed creature on the screen.

If the match is for color, color appears so that the creature matches the other, completed creature on the screen.

Then the computer branches to the Words Procedure.

If there is no match, if the student does not respond within 30 seconds, or if the student presses RETURN without typing, the computer branches to Part B of the Prompting Procedure.

3. What is another question you can ask about how a dupla looks?

<first name> ?

Push RETURN when you have typed another question.

When RETURN is pushed, computer matches for any words for clues (i.e., size, shape, color) that have not yet been matched. If there is a match, graphics change as described above. Then the computer branches to Words Procedure.

If there is no match, if the student does not respond within 30 seconds, or if the student presses RETURN without typing, the computer branches to Part B of the Prompting Procedure.

	Screen	Notes
1.	Sometimes you may want to describe something more complicated, like the setting or a character in a story. Then you must ask questions about the parts rather than the whole thing. Press RETURN.	
2.	Pretend that this rabbit is a character in a story. You want to tell your readers what the rabbit looks like. Press RETURN.	Picture of rabbit. Picture remains on the screen throughout Section C, including branching. Rabbit is white, but it has black ears, feet, and tail.
3.	Now you can ask these question: What shapes are its parts? What sizes are its parts? What colors are its parts? Press RETURN.	Clue graphics on lines with questions.

4. Let's practice. The shape of some parts of the rabbit are important to describe. What shape is the rabbit's ears?

<first name> ?

Clue picture for shape.

When the student pushes RETURN, computer matches for the word <u>pointed</u>. If there is a match, the computer displays the following:

4A Good, *<first name>*. You have some good ideas for describing the shape of the rabbit's ears.

If there is no match, if the student has not typed any words, or if the student presses RETURN without typing, the computer displays the following:

Here is one way to describe the shape of the rabbit's ears: pointed.

4B. Press RETURN.

Computer files student's responses and computer's response (if any) separately for later display and printing for student.

5.

> What shape is the rabbit's tail?
>
> *<first name>* ?

Clue picture for shape.

When the student pushes RETURN, the computer matches for the word round. If there is a match, the computer displays the following:

5A

> Good, *<first name>*. Now you have the idea.

If there is no match, if the student has not typed any words, or if the student pushes RETURN without typing, the computer displays the following:

5B

> Here is one way to describe the shape of the rabbit's tail: round.

Computer files student's responses and computer's response (if any) separately for later display and printing for the student.

33

	The size of some parts of the rabbit are important to describe. Ask some questions about size.
6.	<*first name*> ?

Clue graphic for size.

When the student pushes RETURN, the computer matches for the words feet and ears. If there is a match for both, the computer displays the following:

6A

You are doing well, <*first name*>.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student pushes RETURN without typing, the computer displays the first screen below. If there is no match for feet, the computer displays the second screen. If there is no match for ears, the computer displays the third screen.

6B Screens:

You can ask about the size of the rabbit's ears and feet. Press RETURN.

You can ask about the size of the rabbit's feet. Press RETURN.

You can ask about the size of the rabbit's ears. Press RETURN.

34

The colors of parts of the rabbit are important to describe. Ask some questions about color.	
7.	*<first name>* ?

Clue graphic for color.

When the student pushes RETURN, the computer matches for the words <u>ears</u>, <u>feet</u>, and <u>tail</u>. If there is a match for all three, the computer displays the following:

7A.

Great, *<first name>*. You have the idea now.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student pushes RETURN without typing, the computer displays the first screen below. If there is a match for one of the three, the computer displays the second screen. If there is a match for two, the computer displays the third screen.

7B. Screens:

You can ask about the color of the rabbit's ears, feet, and tail. Press RETURN.

You can also ask about the color of the rabbit's <   > and <   >. Press RETURN.

You can also ask about the color of the rabbit's <   >. Press RETURN.

1. *<Picture of spider>*

Picture of a black spider with long orange and black legs. Its body is large and round. Its eyes are large, green, and diamond shaped. It has violet, v-shaped feet. The picture covers most of the screen. Label: SPIDER.

2. Here is a picture of a spider. What parts of the spider have shapes you would describe?

<first name>

Reduced picture of spider, which remains on screen through rest of Section D; no label for picture.

Clue pictures for shape is on screen.

When RETURN is pushed, computer matches for words <u>feet</u>, <u>eyes</u>, <u>body</u>. If there is a match for all three, the computer displays the following:

2A

> You learn fast, *<first name>*.
>
> Push RETURN.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student pushes RETURN without typing, the computer displays the first screen below. If there is a match for one of the three, the computer displays the second screen. If there is a match for two the computer displays the third screen.

2B Screens:

> You can ask about the shape of the spider's feet, eyes, and body.
>
> Press RETURN.

> You can also ask about the shape of the spider's < > and < >.
>
> Press RETURN.

> You can also ask about the shape of the spider's < >.
>
> Press RETURN.

What parts of the spider have sizes you would describe?

<*first name*> ?

Clue picture for size is on screen.

When RETURN is pushed, computer matches for words <u>legs</u>, <u>body</u>, <u>eyes</u>. If there is a match for all three, the computer displays the following:

3A

You're really good at this, <*first name*>. Press RETURN.

If there is no match, if the student does not begin to type a response within 30 seconds, of if the student pushes RETURN without typing, the computer displays the first screen. If there is a match for one of the three, the computer displays the second screen. If there is a match for two, the computer displays the third screen.

3B Screens:

You can ask about the size of the spider's legs, body, and eyes. Press RETURN.

You can also ask about the size of the spider's < > and < >. Press RETURN.

You can also ask about the size of the spider's < >. Press RETURN.

4. Now, *<first name>*, what parts of the spider have colors you would describe.

<first name> ?

Clue picture for color is on screen.

When RETURN is pushed, the computer matches for words legs, eyes, and feet. If there is a match for all three, the computer displays the following:

4A

> You've got it, *<first name>*!
>
> Press RETURN.

If there is no match, if the student does not begin to type a response within 30 seconds, or if the student pushes RETURN without typing, the computer displays the first screen below. If there is a match for one of the three, the computer displays the second screen. If there is a match for two, the computer displays the third screen.

4B Screens:

> You can ask about the color of the spider's legs, eyes, and feet.
>
> Press RETURN.

> You can also ask about the color of the spider's < > and < >.
>
> Press RETURN.

> You can also ask about the color of the spider's < >.
>
> Press RETURN.

1.	Now you know how to ask yourself these questions in order to get ideas for describing how something looks: /pic. clue/ What shape is it? /pic. clue/ What size is it? /pic. clue/ What color is it? Press RETURN.	Corresponding graphic clues for size, shape, and color appear on the screen on the same line with question.
2.	You also know that sometimes you must ask yourself questions about the parts of the thing you are describing. For example, you might ask, "What is the shape of this thing's parts?" Press RETURN.	
3.	Now you can write a description of the terbie, the dupla, the rabbit, or the spider. Type the name of the one you want to describe. If you want to describe the terbie or the dupla, your ideas will come on the screen. *<first name>* ? Press RETURN.	Pictures of the four objects appear on the screen above their names. If the student does not type in one of the objects, the computer displays the following:

3A

> Please type in TERBIE, DUPLA, RABBIT, or SPIDER. If you do not type in one of these words, I will choose your topic myself.
> *<first name>* ?
>
> Press RETURN when you are finished.

If the student still does not type in one of the topics, the computer displays the following:

3B

> You may write about *<random choice of terbie or dupla>*.
>
> /Picture of the topic./
>
> Press RETURN.

e student selects terble or dupla, the
ter displays screens 4A and 4B.

are your ideas about how a <*chosen*
> looks:

tion: words for each question for
n topic.>
e press RETURN.

Picture of corresponding topic remains on screen. Corresponding clue graphic remains on line with appropriate words.

ou can write your description on paper.
much easier to write a description
you get your ideas first. Push P and
he words will be printed on paper.

luck, <*first name*>.

When P is pushed, computer prints list of words split in two sets. The first set is headed <*first name's*> List and the second is headed Computer's List. Computer then terminates the lesson.

e student selects rabbit or spider, the
ter displays the picture on screen 5.

is a good choice, <*first name*>. Before
rite your description, you will need to
ourself questions about the size, shape,
olor of the <   >'s parts. Push
the questions will be printed on paper.
own your answers on the paper. Then
your description.

luck, <*first name*>.

Picture of chosen topic. When P is pushed, computer prints list of questions with several spaces between each question. Computer then terminates lesson.

DRAFT GRAPHICS

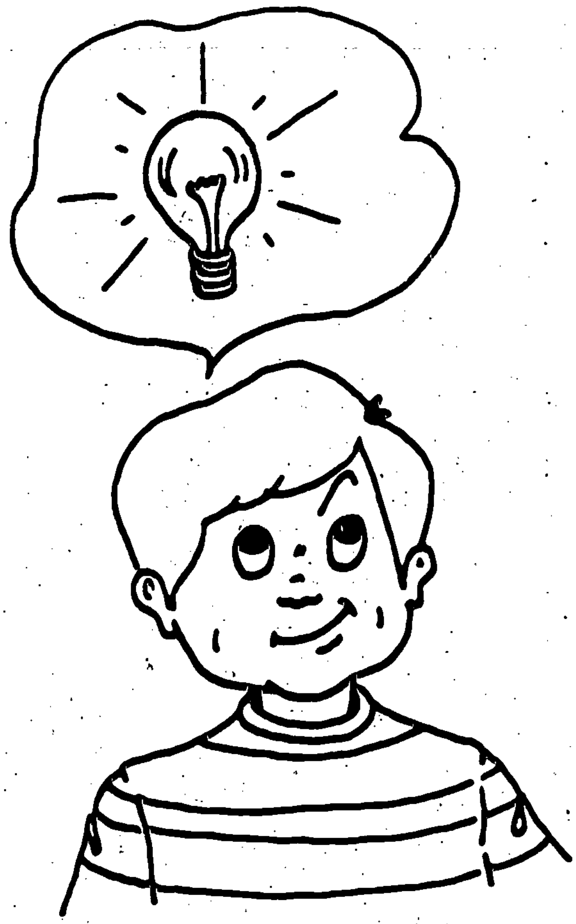
FOR

GENERATING IDEAS

FOR WRITING

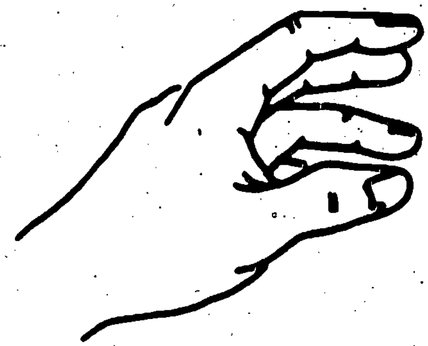
DESCRIPTION

Log-On Procedure



Sense graphics

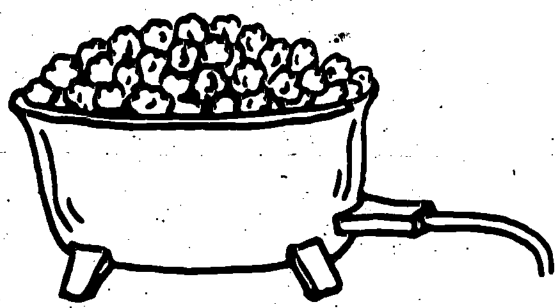
Nose graphic



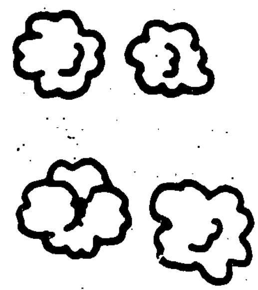
Hand graphic

Mouth graphic

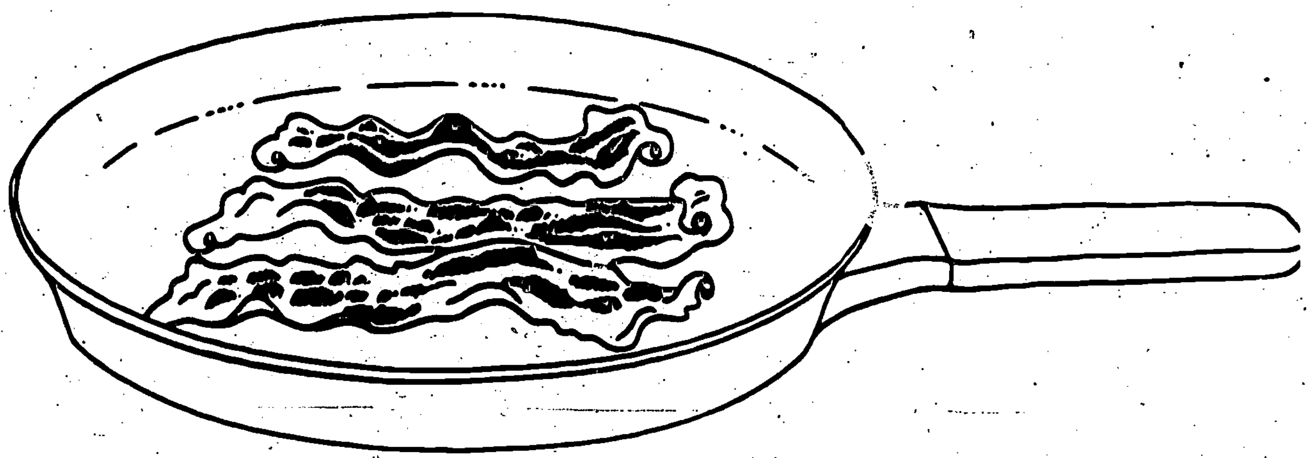
Lesson 1



Popcorn Picture A

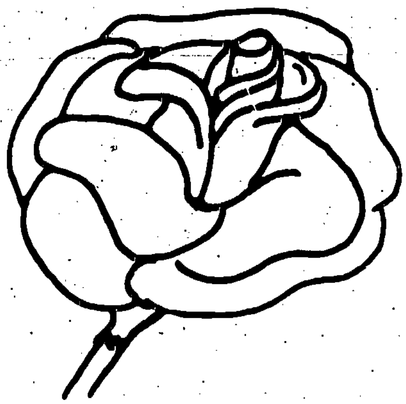


Popcorn Picture B



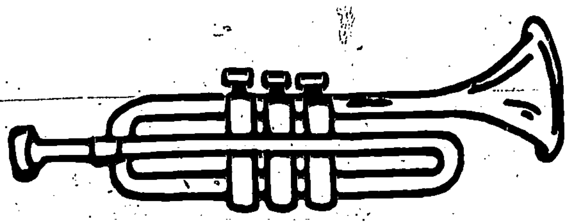
Bacon

Lesson 2

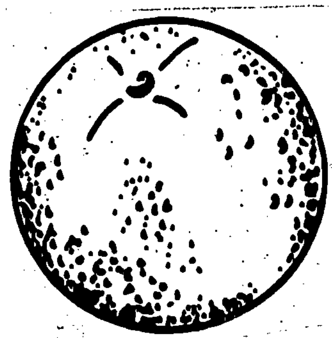


Rose

Student eating rose

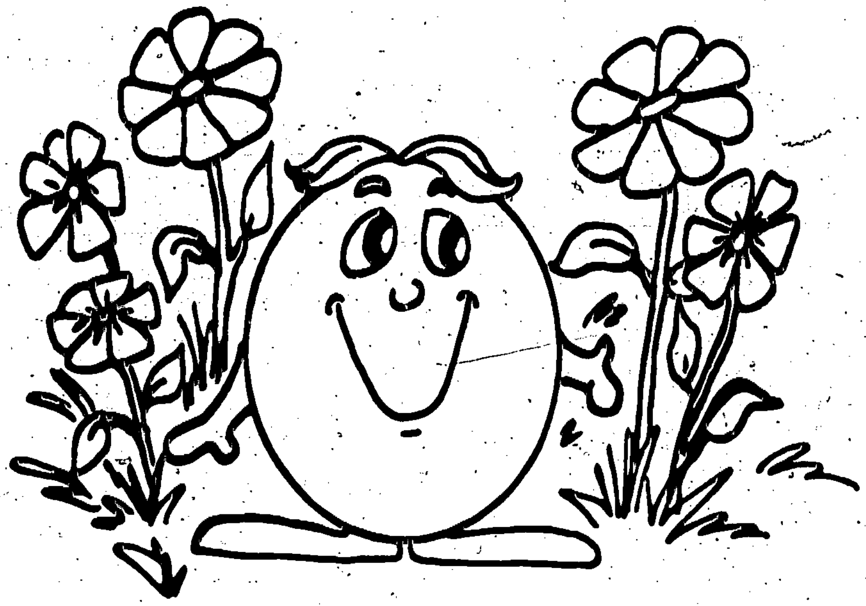


Trumpet

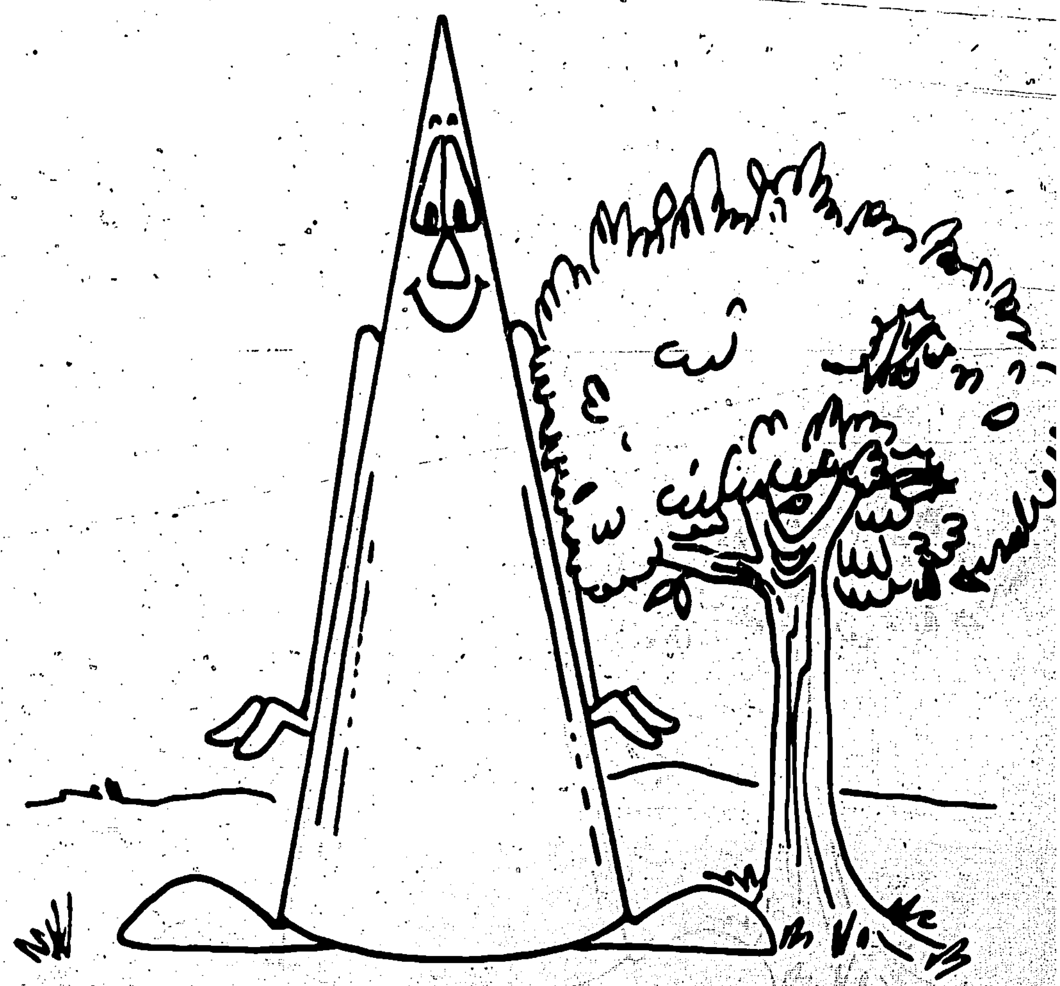


Orange

Lesson 3

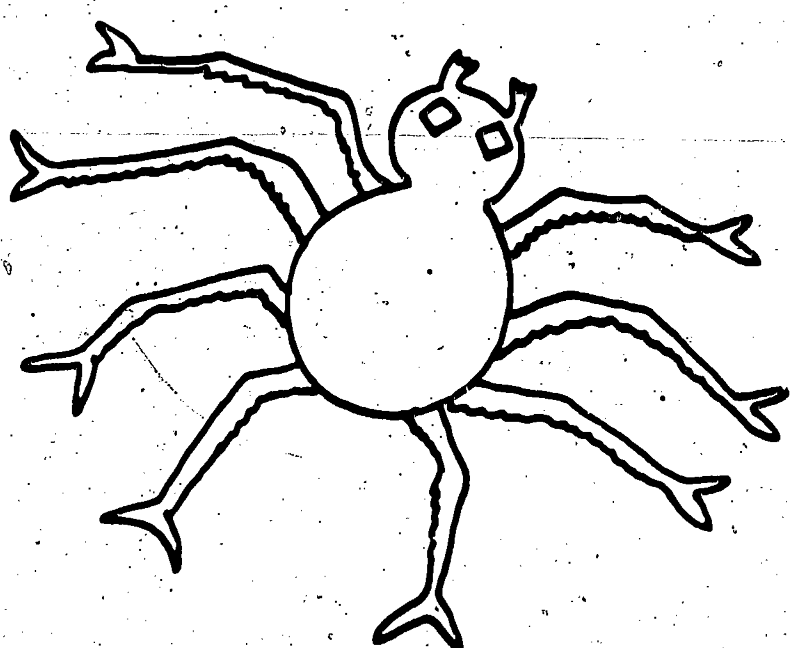


'TERBIE'



"DUPLA"

Rabbit



Spider

COMPUTER INSTRUCTION ON GENERATING IDEAS FOR WRITING DESCRIPTION

SCREEN-DISPLAY SHEETS

On the following pages are screen-display sheets for computer instruction on generating ideas for writing description. These sheets indicate appropriate screen design for the text and graphics. The screens correspond to the descriptions given in the preceding detailed specifications.

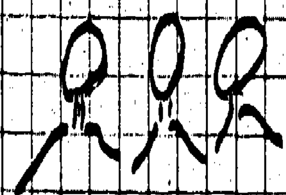
SCREEN: Log-on Procedure 1

COMMENTS

SWRL EDUCATIONAL

RESEARCH AND DEVELOPMENT

PRESENTS



Graphic: Group of children.

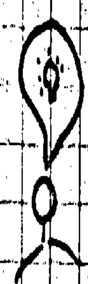
SCREEN: Log-on Procedure 2

COMMENTS

GETTING IDEAS

FOR

WRITING DESCRIPTIONS



SWRL Educational
Research and Development
Los Alamitos, CA

Graphic: One of the same children as in first frame, but with thought balloon with light-bulb flashing off + on above him/her.

SCREEN: Review Procedure 1

COMMENTS

Please type
your first and last names.

Then push the RETURN key.

<   >

I do not know <first and last name>.
Please type the name
you used last time.

Print only if name doesn't match disk file.

<   >

I still do not find the name
<first and last name>.

Please get your teacher
to help me.

Print only if revised name still doesn't match disk file. Teacher finds correct disk and restarts program.

SCREEN: Review Procedure 2

COMMENTS

Hello <first name>

Remember:

- When you see <first name>?
  on the screen, I am asking
  for an answer.
- After you type your answer,
  always push RETURN.
- If you want to stop
  before the lesson is over,
  just type BYE when you see
  your name and a question mark.

Now push RETURN.

SCREEN: Review Procedure 3

COMMENTS

Remember that asking these questions
will help you get ideas for describing
something:

[eye graphic] • What does it look like?

[ear graphic] • What does it sound like?

[nose graphic] • What does it smell like?

[hand graphic] • What does it feel like?

[mouth graphic] • What does it taste like?

Now push RETURN.

Computer goes to first frame of section where student left off in previous section.

SCREEN: Answer Procedure 1

COMMENTS

⟨ ⟩

appropriate sensory graphic

I am waiting for your answer.

⟨first name⟩?

Press RETURN when you are finished.

Question from lesson.

This frame displayed when student doesn't respond to question in lesson.

If a response is now made, branch to say-more procedure.

If no response in 30 seconds, display frame for Answer Procedure 1A.

SCREEN: Answer Procedure 1A

COMMENTS

Here are some words that tell

<question to be answered>:

[appropriate topic graphic]

<words>

Words are filled as part of computers response to question.

Press RETURN.

Computer branches to next frame in Lesson 1, Section B.

SCREEN: Say-More Procedure-1

<appropriate question>

Try to type more words.

[appropriate topic graphic]

[appropriate sensory graphic]

<words already typed by student>

<first name> ?

Push RETURN when you are finished.

COMMENTS

When Return is pushed, computer checks words for match. If all match, branch to next frame in Lesson 1, Section B.

If not all match, branch to Say-more 1A

SCREEN: Say-More Procedure 1A

Here ⟨are some other words/is another word⟩
that tell(s) ⟨what . . . .
. . . . ⟩.

⟨words⟩

Please push RETURN.

COMMENTS

Branch to next frame in Lesson 1, Section B.

SCREEN: PROMPTING PROCEDURE, PART A

COMMENTS

Here is a clue
to a question you can ask:

[appropriate topic graphic] [approp. sensory graphic]

<first name>?

If student response is a match, branch to Word Procedure.

If no match or no response, branch to Prompting Procedure, Part B.

SCREEN: Prompting Procedure 1A

COMMENTS

Here is a question you can ask:

⟨question⟩

[appropriate topic graphic] [approp sensory graphic]

Please push RETURN.

Branch to Words Procedure

SCREEN: Words Procedure

COMMENTS

Now type all the words
you can think of that tell
<appropriate question>.

[appropriate topic graphic] [approp. answer graphic]

<first name>?
Push RETURN when you are finished.

Can you say more, <first name>?
When you are finished, push RETURN.

Print only if student types one or more words. Branch to proper frame in lesson (see specifications memo).

Here is a word that tells
<appropriate question>.

<word>

Print only if student does not type one or more words. Branch to proper frame in lesson (see specifications memo).

SCREEN: Termination Procedure 1

COMMENTS

That is enough for now.

We will talk another time
about getting ideas
for writing descriptions.

Goodbye, <first name>.

Print if operating time is close to 20 minutes and computer is about to begin a new topic.

SCREEN: Termination Procedure 2

COMMENTS

Goodbye <first name>.

We will work later on getting ideas for writing descriptions.

Displayed if student types BYE while computer is waiting for input.

SCREEN: Termination Procedure 3A

If you have finished,
please push the RETURN key.

If you do not, I must say goodbye.

COMMENTS

Print only if student responds to a command or a question, but does not push RETURN key within 30 seconds.

If RETURN is pushed, continue lesson

If RETURN is not pushed within 15 seconds, go to Termination Procedure 3B.

SCREEN: Termination Procedure 3B

COMMENTS

I'm sorry. Goodbye <first name>.

Print only if student does not respond to Termination Procedure 3A.

SCREEN: LESSON 1: SECTION A (INTRODUCTION) 1

COMMENTS

Please type
your first and last names.

Be sure you type correctly.

Then push the RETURN key.

SCREEN: LESSON 1: SECTION A (INTRODUCTION) 2

COMMENTS

Hi, <first name>.

Whenever you see <first name>?
on the screen,
I am asking for an answer.

After you answer,
always push the RETURN key.

Let's practice.
What grade are you in?

<first name> ?

SCREEN: LESSON 1: SECTION A (INTRODUCTION) 3

COMMENTS

Good, {first name}.
Before we begin,
I will tell you how to escape!

If you want to stop at any time
during the lesson, just type BYE
when you see your name
and a question mark.
I'll get the message.

Push RETURN.

SCREEN: LESSON 1; SECTION B (POPCORN) 1

COMMENTS

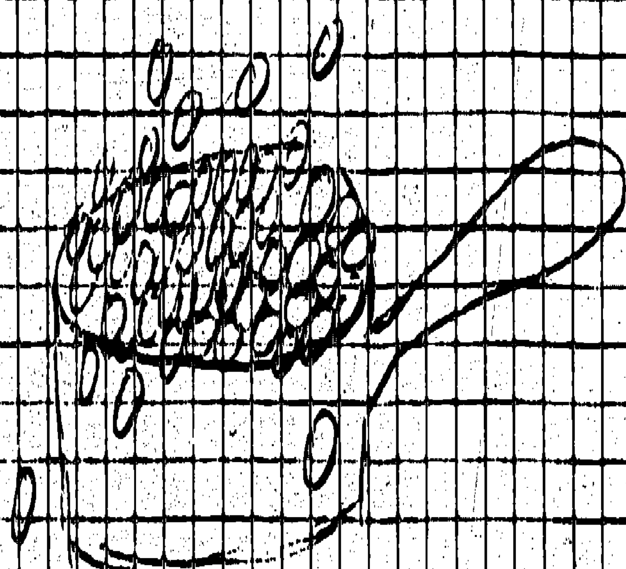
Now you are going to learn
how to get ideas
for writing descriptions.

Then you will use your ideas
to write a description.

Graphic: Student with thought-balloon; Light-bulb inside (sa... Graphic as in Log-On Procedure

SCREEN: LESSON 1: SECTION B (POPCORN) 2

Here is a picture of popcorn.



POPCORN

COMMENTS

Graphic:
Popcorn popping; popping sounds occur simultaneously with explosion of kernels on screen.

Delay before next screen.

SCREEN: LESSON 1: SECTION B (POPCORN) 3

COMMENTS

Here is another picture of popcorn.

[individual pieces of popcorn]

Graphic: Several individual pieces of popcorn.

Delay before next screen

SCREEN: LESSON 1: SECTION B (POPCORN) 4

Now let's descibe popcorn.
What does popcorn look like?
Type all the words you can think of
that tell what popcorn looks like.
Then push the RETURN key.

[popcorn graphic] [eye graphic]

<first name> ?

COMMENTS

Popcorn graphic from last frame remains on screen.

If Return is pushed but no words have been typed, branch to Answer Procedure.

If no response in 30 seconds, branch to Answer Procedure.

If student types one or more words and pushes RETURN, branch to SAY-MORE Procedure.

SCREEN: LESSON 1: SECTION B (POPCORN) 5

COMMENTS

What does popcorn sound like?
Type all the words you can think of
that tell how popcorn sounds
when it is popping.

[popcorn graphic] [ear graphic]

<first name>?

Popcorn graphic is same as in Frame 4.

If no response in 30 seconds or Return is pushed without words being typed, branch to Answer Procedure.

If student types one or more words and pushes RETURN, branch to SAYMORE Procedure.

SCREEN: LESSON 1: SECTION A (POPCORN) 6

COMMENTS

What does popcorn smell like?
Type all the words you can think of
that tell how popcorn smells.

[popcorn graphic] [nose graphic]

Popcorn graphic is same as in Frame 5.

<first name>?

If no response in 30 seconds, or if RETURN is pushed without words being typed, branch to Answer Procedure.

If student types one or more words and pushes RETURN, branch to Say-more Procedure.

hat does popcorn feel like
hen you touch it?
ype all the words you can think of
hat tell how popcorn feels
hen you pick it up in your hand.

[popcorn graphic] [hand graphic]

irst name)?

COMMENTS

Popcorn graphic is same as in Frame 6.

If no response in 30 seconds or if words are typed without RETURN being pushed, branch to Answer Procedure.

If student types one or more words and pushes RETURN, branch to Say More Procedure.

SCREEN: LESSON 1; SECTION A (POPCORN) 8

COMMENTS

What does popcorn taste like?
Type all the words you can think of
that tell how popcorn tastes.

[popcorn graphic] [mark graphic]

Popcorn graphic is same as in Frame 7.

⟨first name⟩ ?

If no response in 30 seconds, or if RETURN is pressed without words being typed, branch to Answer Procedure.

If student types one or more words and pushes RETURN, branch to Say-More Procedure.

SCREEN: Lesson 1: Section B (Popcorn) 9

COMMENTS

[popcorn graphic]

Same popcorn graphic as Frame 8, but rolled up.

When we asked questions
about what popcorn looks,
sounds, smells, feels,
and tastes like,
we got these words:

[eye graphic] <words for looks>

Same eye graphic as Frame 4, but moved.

Push RETURN.

SCREEN: Lesson 1: Section B (Popcorn) 10

[popcorn graphic]

[ear graphic] <words for sounds>

[nose graphic] <words for smells>

[hand graphic] <words for feels>

[mouth graphic] <words for tastes>

Push RETURN.

COMMENTS

Same popcorn graphic as in Frame 9.

Same sensory graphics as in Frames 5-8, but moved as indicated.

SCREEN: Lesson 1: Section C (Bacon) 1

COMMENTS

[bacon graphic]

Here is a picture of bacon.

[nose graphic] [hand graphic] [eye graphic] [ear graphic] [mouth graphic]

The other pictures give clues to the questions you can ask.

What is one question you can ask to get ideas for writing?

<first name>?

Push RETURN when you have asked ONE question.

<response>

If student response matches a sensory cue, corresponding cue disappears. Branch to What Procedure

If no match in 30 seconds or if student pushes RETURN w/out typing, branch to Part B of Prompting Procedure.

SCREEN: Lesson 1: Section C (Bacon) 2

COMMENTS

[bacon graphic]

Bacon graphic: Same as in Frame 1

[nose graphic] [hand graphic] [eye graphic] [ear graphic] [mouth graphic]

Print only sensory cues that remain after Frame 1.

What is one other question
you can ask?
Remember to look at the clues.

<first name> ?

Press RETURN when you have typed
another question.

If student response matches a sensory cue, appropriate graphic disappears. Branch to Word Procedure.

If no match or no response in 30 seconds, or if RETURN is pushed without typing, branch to Part B of Prompting Procedure.

SCREEN: LESSON 1: SECTION C (BACON) 3

bacon graphic

Now let's look at the ideas
for writing a description of bacon.

When we asked questions
about how bacon looks, sounds,
smells, feels, and tastes,
we got these words:

eye graphic

<words for looks>

Push RETURN to continue.

COMMENTS

Graphic: Same as Frame 2

SCREEN: Lesson 1: Section C (Bacon) 4

COMMENTS

[bacon graphic]

[ear graphic] ⟨words for sounds⟩

[nose graphic] ⟨words for smells⟩

[hand graphic] ⟨words for feels⟩

[mouth graphic] ⟨words for tastes⟩

Push RETURN.

Same graphic cues as shown on Lesson 1: Section B (Popcorn) Frame 10.

SCREEN: LESSON 1; SECTION D - 1

COMMENTS

Now you can write a description
of popcorn or bacon.

Type the name of the one
you want to describe.
Your ideas will come on the screen.

<Print name>?

If student types in topic, proceed to D-2.

If student does not respond in 30 sec display D-1A.

SCREEN: LESSON 1: SECTION D-1A

COMMENTS

Please type in popcorn or bacon.
If you do not type in
one of these words,
I will choose your topic.

Press RETURN when you are finished.

<first name>?

If student types in topic, proceed to D-2.

If student does not respond in 30 sec, display D-1B.

LESSON 1: SECTION D-1B

COMMENTS

You may write on <random choice of the two topics>.

[Picture of topic]

Press RETURN.

SCREEN: Lesson 1: Section D-2

COMMENTS

Here are the ideas, <first name>:

[eye graphic] <words for looks>

[ear graphic] <words for sounds>

[nose graphic] <words for smells>

[hand graphic] <words for feels>

[mouth graphic] <words for tastes>

Please press RETURN.

SCREEN: LESSON 1: SECTION D-3

COMMENTS

Now write your description on paper,
using the ideas you have for writing.
Push P and all the ideas
will be printed on paper.

Have fun, (first name).

SCREEN: LESSON 2: SECTION A (ROSE) 1

COMMENTS

[mouth graphic] [hand graphic] [eye graphic] [nose graphic] [ear graphic]

Sometimes you don't want to or can't
ask all five questions about
the thing you are describing.

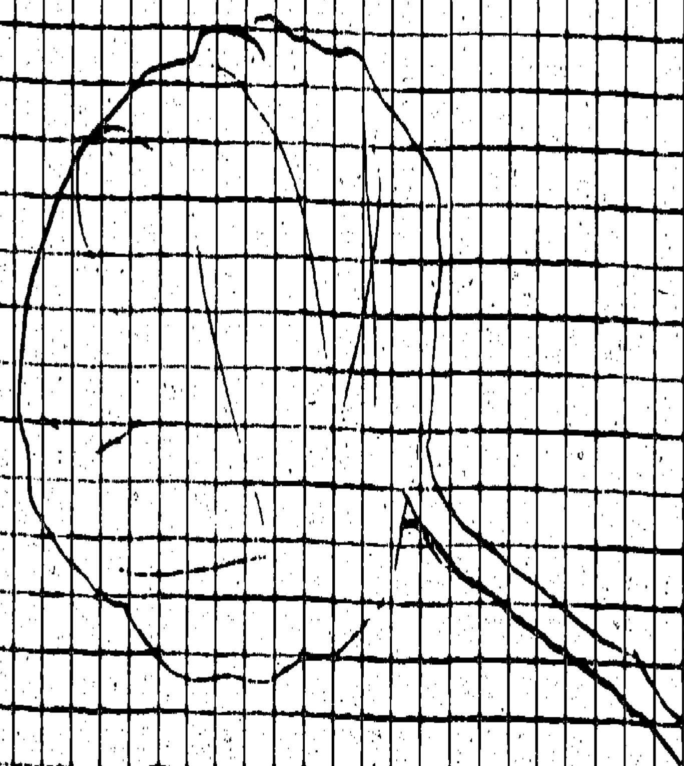
Please press RETURN.

5 sense cue graphics on screen. All but eye-graphic disappear one at a time. Eye graphic leaves screen when RETURN is pushed

SCREEN: LESSON 2: SECTION A (ROSE) 2

COMMENTS

Here is a rose.



ROSE

Picture of rose fills center of screen

SCREEN: LESSON 2: SECTION A (ROSE) 3

[student eating rose]

You would not ask
what this rose tastes like
unless you are very strange
and eat roses.

COMMENTS

Picture of student in log-on procedure about to take a bite from a rose (no animation).

SCREEN: LESSON 2: SECTION A (ROSE) 4

[student listening to rose]

You would not ask
how the rose sounds.

Press RETURN.

COMMENTS

Picture of same student listening to a rose, musical notes on the screen.

SCREEN: LESSON 2: SECTION A (ROSE) 5

COMMENTS

[rose]

What is one question you would ask?

<first name> ?

Press RETURN when you have written ONE question.

Reduced picture of rose as in Frame 2. (no label for reduced picture)

If keyword matches, branch to Words Procedure.

If no match or no response in 30 sec, or if student pushes RETURN without typing, branch to Prompting Procedure.

SCREEN: LESSON 2: SECTION A (ROSE) 6

rose

What is another question
you would ask about this rose
to get ideas for describing it?

<first name>?

Press RETURN when you have written
another question.

COMMENTS

Picture of rose
Same as Frame 5

If unused key words match branch to Words Procedure.

If no match or no response in 30 sec, or if student pushes RETURN without typing, branch to Prompting Procedure.

SCREEN: LESSON 2: SECTION B (TRUMPET) 1

COMMENTS

Trumpet

TRUMPET

Here is another picture
to ask questions about,
but you wouldn't ask
how this trumpet smells.

Press RETURN.

Picture of trumpet & label

SCREEN: LESSON 2: SECTION B (TRUMPET) 2

COMMENTS

trumpet

What is one question you would ask about this trumpet to get ideas for describing it?

<first name> ?

When you have typed ONE question, press RETURN.

Same picture of trumpet as in Frame 1. No label

If key words match, branch to Words Procedure.

If no match or no response in 30 sec., or if student presses RETURN without typing, branch to Prompting Procedure.

SCREEN: LESSON 2: SECTION B (TRUMPET) 3

Trumpet

What is another question
you would ask about this trumpet
to get ideas for describing it?

<first name>?

Push RETURN when you have typed
another question.

COMMENTS

Same picture as Frame 2.

If unused key words match, branch to Words Procedure.

If no match or no response in 30 sec., or if student pushes RETURN without typing, branch to Prompting Procedure.

SCREEN: LESSON 2: SECTION C (ORANGE) 1

COMMENTS

orange

ORANGE

Here is one more picture
to ask questions about.

Please press RETURN.

Picture of orange
and label

SCREEN: LESSON 2: SECTION C (ORANGE) 2

COMMENTS

orange

Same picture as Frame 1, no label.

What is one question you would ask
about this orange to get ideas
for describing it?

If key words match, branch to Words Procedure.

<first name>?

Please push RETURN
when you have typed ONE question.

If no match, or no response in 30 sec., or if student presses return without typing, branch to Prompting Procedure.

SCREEN: LESSON 2: SECTION C (ORANGE) 3

COMMENTS

[orange]

What is another question
you would ask about this orange
to get ideas for describing it?

(first name)?

Please push RETURN
when you have typed another question.

Same picture as
Frame 2

If unused key
words match,
branch to Words
Procedure.

If no match or
no response in
30 sec., or if
student pushes
RETURN without
typing, branch to
Prompting Procedure.

SCREEN: LESSON 2: SECTION D (Compose) 1

COMMENTS

Now you know how to ask yourself these questions in order to get ideas for writing descriptions:

[<eye>] What does it look like?

What does it sound like? [<ear>]

[<nose>] What does it smell like?

What does it feel like? [<hand>]

[ ] What does it taste like?

Press RETURN.

Close graphics on same line as questions.

SCREEN: Lesson 2: Section D (Compose) 2

Now you can write a description
of a rose, a trumpet, or an orange.

Type the name of the one
you want to describe.
Your ideas will come on the screen.

<first name>?

Press RETURN when you are finished.

COMMENTS

If student selects a topic, corresponding picture comes on screen.

If student does not type in one of the topics, display 2A.

SCREEN: LESSON 2: SECTION D (COMPOSE) 2A

Please type in
rose, trumpet, or orange.
If you do not type in
one of these words,
I will choose your topic.

Press RETURN when you are finished.

<first name>?

COMMENTS

If student does not enter topic, display 2B.

When student selects topic, picture of [illegible] comes on screen.

SCREEN: LESSON 2: SECTION D (COMPOSE) 2B

COMMENTS

You may write about
<randomly selected topic>.

[picture of selected topic]

Please press RETURN.

: Lesson 2: Section D (Compose) 3

COMMENTS

Here are the ideas, <first name>. [picture of selected topic]

[eye] <words for topic>

[ear] < " " " >

[nose] < " " " >

[hand] < " " >

[mouth] < " " " >

Please press RETURN.

appropriate cues and words that correspond with selected topic

SCREEN: Lesson 2: Section D (Compose) 4

COMMENTS

Now you can write
your description on paper.

[picture of selected topic]

You will see how easy it is
to write when you get
your ideas first.

Push P and all the words
will be printed on paper.

Good luck, <first name>.

SCREEN: LESSON 3; SECTION A (TERBIE) 1

COMMENTS

picture of "Terbie"

TERBIE

SCREEN: LESSON 3: SECTION A (TERBIE) 2

COMMENTS

Sometimes
when you describe something,
you can ask yourself
only one question:

What does it look like? 

Please push RETURN.

Eye graphic on same line as question.

SCREEN: LESSON 3: SECTION A (TERBIE) 3

Here is an imaginary animal called a terbie.

When you want to describe what this terbie looks like you can ask yourself these questions:

What shape is it?

What size is it?

What color is it?

Press RETURN.

Terbie

COMMENTS

graphic: fat and thin stick figures

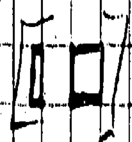
graphic: tall and short stick figures

graphic: same size & shape - different colors.

Questions appear one at a time after a pause and remain on screen. Clue pictures appear with corresponding questions.

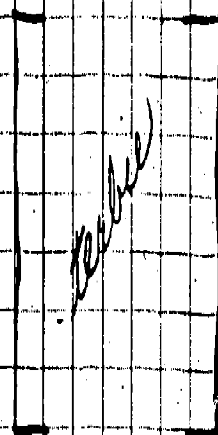
SCREEN: LESSON 3: SECTION A (TERBIE) 4

Let's practice.
Write all the words
you can think of
that answer the question:

What shape
is the terbie? 

{first name}?

Press RETURN when you are finished.



COMMENTS

graphic 2: stick figures for Pat + Kim.

SCREEN: LESSON 3: SECTION A (TERBIE) 4A

COMMENTS

Good, <first name>.
You have some good ideas
for descibing the shape
of a terbie.

Press RETURN.

Terbie

SCREEN: LESSON 3: SECTION A (TERBIE) 4B

COMMENTS

Here is one way to describe
the shape of the terbie:

round

terbie

Press RETURN.

sound flashes off and on.

SCREEN: LESSON 3: SECTION A (TERBIE) 5

COMMENTS

Now answer this question:

What size
is the terbie?

<first name> ?

Terbie

Press RETURN when you are finished.

Graphic: stick figures for tall + short.

SCREEN: LESSON 3; SECTION A (TERBIE) 5A

COMMENTS

You are doing well, <first name>.
You have some good ideas
for describing the size
of the terbie.

Press RETURN.

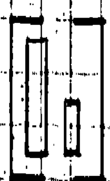
[terbie]

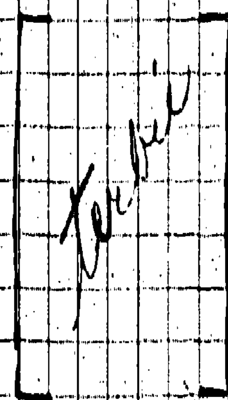
293

SCREEN: LESSON 3; SECTION A (TERBIE) 5B

COMMENTS

Here is a good way
to describe the size
of a terbie:

small 



Press RETURN.

small flashes
off on on

SCREEN: LESSON 3. SECTION A (TERBIE) 6

COMMENTS

Here is one more question:

What color
is the terbie? □□

terbie

<first name> ?

Press RETURN when you are finished.

Graphic 2: stick figures for color

If student enters golden or yellow display 6A

If no match or no response in 30 sec or student pushes RETURN without typing, display 6B

SCREEN: LESSON 3: SECTION A (TERBIE) 6A

COMMENTS

Great, <first name>!
Now let's do
something different.

Press RETURN.

Terbie

SCREEN: Lesson 3: Section A (Terbie) 6B

Here is a good way
to describe the color
of a terbie:

<color word>

terbie

Now let's do something different.

color wo...
off ...

SCREEN: LESSON 3: SECTION B (DUPLA) 1

COMMENTS

[dupla]

[outline of dupla]

Here is a picture
of an imaginary animal
called a dupla.

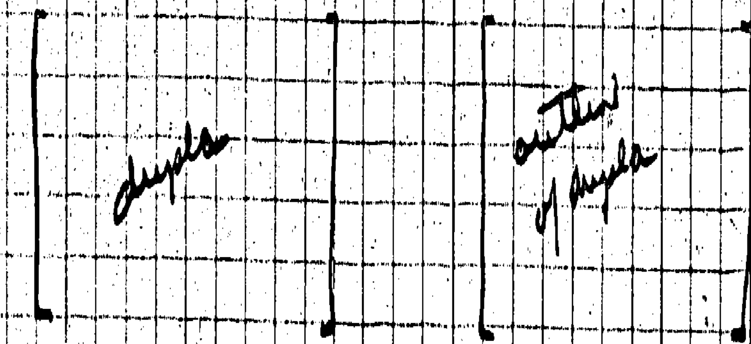
The other is also
supposed to be a dupla,
but it doesn't look like one now.

It will become a dupla
as you tell how to describe
the first dupla.

Press RETURN.

SCREEN: LESSON 3: SECTION B (DUPLA) 2

COMMENTS



What is one question
you can ask about how the dupla looks?
<first name>?

Press RETURN
when you have typed one question.

SCREEN: LESSON 3: SECTION 8 (DUPLA) 9

COMMENTS

What is another question you can ask about how a dupla looks?

<first name> ?

Push RETURN when you have typed another question.

Graphics: Same as at end of frame 2, with partially completed dupla in 2nd pic.

If student enters unused words for size, shape, or color, graphics change as in frame 1. Then branch to Words Procedure.

If no match, or no response in 30 sec, or student enters RETURN without typing, branch to Part B of Prompting Procedure.

Lesson 3: Section C (Rabbit)

--screens similar to those in preceding sections

Lesson 3: Section D (Spider)

--screens similar to those in preceding sections

SCREEN: LESSON 3: SECTION E (COMPOSE) 1

Now you know how to ask yourself these questions in order to get ideas for describing how something looks:

What shape is it?

What size is it?

What color is it?

Press RETURN.

COMMENTS

Graphic clues appear on screen on same line as question.

SCREEN: LESSON 3: SECTION E (COMPOSE) 2

COMMENTS

You also know that sometimes
you must ask yourself questions
about the parts of the thing
you are describing.

For example,
you might ask,
"What is the shape
of this thing's parts?"

Press RETURN.

SCREEN: LESSON 3: SECTION E (COMPOSE) 3

COMMENTS

terbie dupla rabbit spider

Now you can write a description
of the terbie, the dupla,
the rabbit, or the spider.
Type in the name of the one
you want to describe.
If you want to describe the terbie
or the dupla, your ideas will come
on the screen.

<first name>?

Press RETURN.

If student does not type in one of the topics, display 3A.

SCREEN: LESSON 3: SECTION E (COMPOSE) 3A

COMMENTS

Please type in
TERBIE, DUPLA, RABBIT, or SPIDER.
If you do not type in
one of these words,
I will choose
your topic myself.

<first name>?

Press RETURN when you are finished.

If student does not enter a topic, display 3B.

SCREEN: LESSON 3: SECTION E (COMPOSE) 3B

COMMENTS

You may write about <   >.

Press RETURN.

SCREEN: LESSON 3: SECTION E (COMPOSE) 4A

COMMENTS

Here are your ideas
about how a <topic> looks:

< questions/words >

Please press RETURN.

LESSON 3: SECTION E (COMPOSE) 4B

COMMENTS

ow you can write
our description on paper.
t is much easier
o write a description
hen you get your ideas first.
ush P, and all the words
ill be printed on paper.

ood luck, <first name>.

SCREEN: LESSON 3: SECTION E (COMPOSE) 5

COMMENTS

That is a good choice,
<first name>.

Before you write your
description, you will need to
ask yourself questions about
the size, shape, and color of
the <   >'s parts.

Picture of chosen topic

Push P and the questions will
be printed on paper.

Jot down your answers on the paper.
Then write your description.

Good luck, <first name>.

Narrative Text

SOUTHWEST REGIONAL LABORATORY
WORKING PAPER

DATE: July 1, 1982
NO: WP 2-82/02

COMPUTER INSTRUCTION FOR GENERATING AND REVISING/EDITING NARRATIVE TEXT

Bruce Cronnell

ABSTRACT

General specifications are provided for computer-based instruction in generating and revising/editing stories. The student is presented with a filmed story to write about and is provided assistance while composing on a word-processor. When the student has completed the story, the computer analyzes the story and interacts with the student to help her/him revise and edit the story.

COMPUTER INSTRUCTION FOR GENERATING AND REVISING/EDITING NARRATIVE TEXT

Bruce Cronnell

One goal of computer-based instruction in composition is to have the computer interact with students while they are composing text on a word processor. The computer could provide the stimulus for writing, assist the student while he/she is writing, and help the student to revise and edit the text. This paper outlines a procedure to do these tasks for the writing of narration. The scenario contains many operational details, and additional specifications needed to make the application operational are also noted in the paper.

Basic Procedures

Before students could begin using this proposed instructional module, they would have to learn how to use a specialized word processor. Word-processor instruction would be both off-line and on-line. The particular word processor used in instruction is not specified in this paper; it would not be one commercially available, but one specially developed for instructional purposes. There are two reasons for this. First, the word-processor software must share the same disk as the instructional software so that the computer-student interaction can take place efficiently during the writing process. Secondly, most commerical word-processing programs are more sophisticated/complex than is needed for student writing.

The word processor would have several basic requirements: upper- and lower-case letters, word wrapping at line breaks, scrolling. Moreover, its function keys (e.g., for capitalization, deletion, insertion) would be straightforward and easy to use. Other characteristics of a word processor optionally appropriate for instructional use can be specified later; however, all such features are technically straightforward design matters.

When a student is ready to use this narrative-text program, she/he would insert two disks into the disk drives: the program disk and the individual student disk, on which text produced would be recorded.

The computer would first display certain title screens and then a menu. The menu would list a series of story titles, and the student could select the story that she/he wished to write about. Only a single story is exemplified in this paper, one based on pictures available in SWRL files and consequently used for illustrative convenience. The story is obviously intended for elementary-school students; more sophisticated stories could also be used. The basic computer procedures suggested here would be used with all stories, with variations in wording--and in illustrations--inserted at the appropriate places depending on the story chosen by the student.

If the student has not used the program before (as evidenced by a blank student disk), the computer introduces the operation of the program. If the student has used the program before, the computer reviews the operation of the program and sends the student to where she/he left off the last time when in the midst of a story.

The following are critical operational procedures in the program:

1. When the student's first name followed by a question mark is displayed, the student is expected to make a response.
2. After the student has made a response, she/he must push the RETURN key.
3. Four special function keys (referred to as W, X, Y, and Z in this paper) must be used to signal specific functions to the computer:

   W = to exit the program

   X = to ask for help

   Y = to indicate completion of writing

   Z = to scroll text

   Ideally, a labeled key would be used for each function (i.e., BYE, HELP, DONE, SCROLL). However, the keyboards for some computers (e.g., the Apple II) do not have enough extra keys to permit this. Therefore, the solution may be to have the student push a function key (but not the one used for upper-case letters) and then type a word (e.g., "bye") or a letter, chosen for mnemonic value (e.g., "E" for exit/end.)

<u>Writing</u>

The student is introduced to the content of the story to be written--the characters and perhaps the setting (and maybe even something about the story content). Then a filmed version of the story is presented. The film would be displayed from either videotape or videodisc and would be displayed either on the screen on which text is displayed or on a separate monitor.

The film can be either animated or live-action. It can present either a complete story for students to write or an open-ended story for students to complete. After the film is displayed, the student is told to write her/his story. While the student is writing, one or more still pictures from the story remains on the screen (whatever screen the film was displayed on) to remind the student of story content.

See Figure 1 for a sample of the displays that might be shown at this stage.

At the top of the screen where the student is to enter text, the title of the story is given (to ensure that the story is titled and to help remind the student of story content). The title is the beginning of the student's text; it is part of the student's text and scrolls off the top of the screen (along with text) when text fills up the screen.

Help

The student receives help in one of two ways:

1. When the student pushes the X function key, the help menu is presented, either in a "window" on the screen or on a screen that replaces the student text.
2. When the student stops typing for 60 seconds, a message is displayed at the bottom of the screen asking if the student wants some help.
    a. If the student responds "yes," the help menu is presented.
    b. If the student responds "no," the question message is replaced briefly with the message "WRITE SOME MORE."

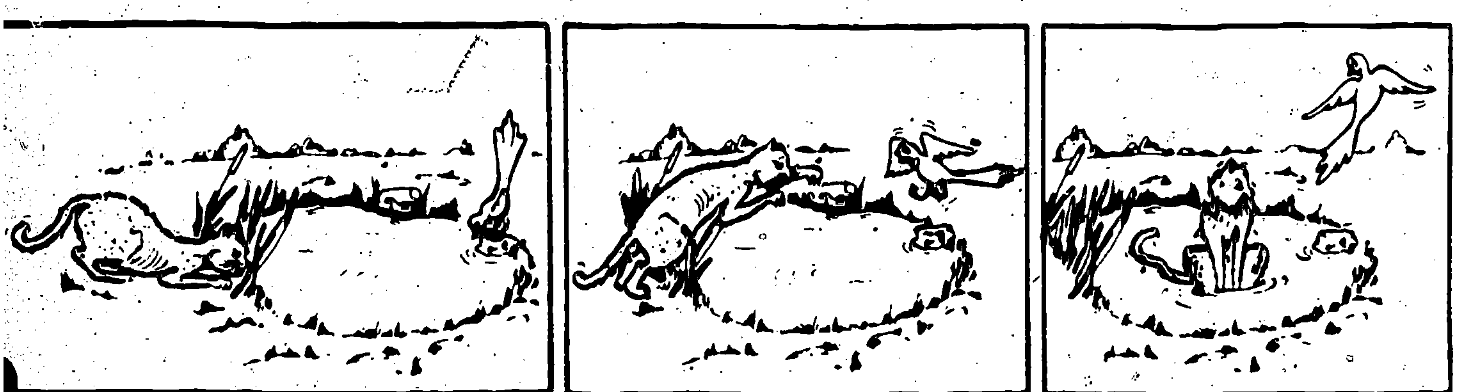
1. | Today you will write a story.

[picture of cat]
[picture of bird]

The story is about a cat and a bird.

Watch to see what the cat and the bird do.

2. Display animated version of the story suggested by the following pictures.*



3. | Now write a story about the cat and the bird.

Tell what the cat does and what the bird does.

Figure 1. Sample Displays to Introduce Writing Task

*During student writing, these three pictures would be displayed.

The help menu asks the student what kind of help she/he wants:

1. to see the film again
2. to figure out what to write next
3. to see some words to use in the story
4. to see what she/he has written
5. to stop writing

The student responds by typing in the number of the help wanted, and the computer responds appropriately:

1. The filmed version of the story is presented (as described above).
2. Questions are displayed to encourage writing. For example, for the sample story used in this paper:

   What does the cat do?

   What does the bird do?

   Where does the story happen?
3. Words are displayed that might be used in the story. For example, for the sample story:

cat	bird	next
water	pond	then
catch	hide	after
fall	wet	before
4. The student text is displayed from the beginning. (The student is instructed to use the Z function key to scroll additional text onto the screen.)
5. A message is displayed asking the student whether she/he has finished writing her/his story. If yes, the program moves to "Revising/Editing" (see below); if no, the program returns to the student text.

After each of these helps is shown (except after a "yes" response to number 5), the computer returns to the student text, displaying briefly the message "WRITE SOME MORE."

Revising/Editing

When the student pushes the Y function key or when the student answers "yes" to question 5 of "Help," the student is assumed to have completed her/his first draft. Although the student may have made changes while writing the text, the next part of the program focuses specifically on revising and editing.*

Research suggests that the first step in changing text is to make revisions--that is, make changes in substance. Then the text is edited--that is, surface features (e.g., spelling, punctuation, capitalization) are corrected. While this order is practical for ordinary writing, it does not work well for computer interaction with the changing process. Since the computer reads text literally, errors must be corrected first so that the computer can make more "intelligent" suggestions for more complicated revisions.

In the first step of the "Revising/Editing" procedure, the student is told to read her/his story again and to correct any mistakes and make any changes that will make the story better. (Ideally, the student would have done this before exiting the writing part of the program. However,

*The revision and editing routines outlined here are only a few of many that could be incorporated into a computer instructional program. The exact changes to be treated in the program would depend on many factors, such as grade level, story content, and relation to ongoing classroom instruction in composition this step ensures that students do in fact re-read their texts.) The student text is displayed from the beginning; the student scrolls through it, making any changes and corrections. When the student has gone through her/his text, the program returns to the remainder of the "Revising/Editing" steps.

The second step is a computer check of the spellings in the student text. The computer has a small dictionary in memory; i.e., a word list of 3,000-5,000 words, including all words expected in the story. (There can be space limitations if a dictionary is included in the instructional software; if so, the dictionary could be on a separate disk that the student is directed to use.) The computer matches words in the student text against words in the dictionary, ignoring capitalization and accepting forms with inflectional suffixes. For each text word not found in the dictionary, the sentence containing the word is displayed with the unmatched word underlined. A message is also displayed:

> You have a word that I don't understand.
>
> Please check the underlined word to be sure that it is spelled right.
>
> Correct the spelling of the word and push RETURN. If the word is right, just push RETURN.

Note that the student is allowed to continue without a correction if she/he thinks the word is correct, because the unmatched word may in fact be a correct word that is not in the dictionary. Note also that this routine does not ensure that all spellings are appropriate; for example, incorrectly used homophones would be treated as correct spellings.

Next the computer searches for a capital letter at the beginning of the text and after each terminal punctuation mark (i.e., period, question mark, exclamation mark). For each missing capital letter, the computer displays the sentence needing capitalization, along with the preceding sentence, and asks the student to add a capital letter. If the student does not respond or responds incorrectly, the computer performs the capitalization and graphically highlights it.*

A similar procedure is used to check for lack of terminal punctuation and to have the student add the appropriate mark. The computer searches for terminal punctuation at the end of the text and before each capital letter, with the following exceptions: the first word in the text; the titles <u>Mr</u>., <u>Mrs</u>., <u>Ms</u>., <u>Dr</u>., <u>Miss</u>; words that are likely to be names in the story (e.g., in the sample story: "Bird," "Cat," or "Pond"). However, students may misuse capital letters by placing them within sentences or they may use proper nouns within sentences. Consequently, so that inappropriate periods are not added, the computer does not automatically add a period if the student does not make a correction (as opposed to the automatic capitalization provided for above).

The next two steps are designed to help students improve their sentence structure: to combine short sentences or to break up long sentences.** First the computer searches for any consecutive sequence of

---

*This procedure will add inappropriate capital letters if abbreviations are used in the text. However, abbreviations are generally not acceptable in text (except for titles, which are followed by capitalized names).

**The word length to define short and long sentences may vary depending on grade level; the numbers used here are appropriate for elementary school.

two sentences, both shorter than five words. The pair of short sentences is displayed, with the notation that they are short and the question "Can you turn them into one sentence?" Next the computer searches for any sentence longer than 20 words. The long sentence is displayed, with the notation that it is long and the question "Can you change this into two shorter sentences?"

The following step is designed to improve sentence variety. The computer searches for any two (or more) consecutive sentences beginning with the same word or, in the case of sentences that begin with "A," "An," or "The," beginning with the same two words. If such sentences are found, they are displayed, with the notation that they begin the same way and the question "Can you change one sentence to make the beginnings different?"

The computer then searches, one at a time, for specific words that would be expected in the story the student has written; e.g., in the sample story: bird, cat, pond/water, next/then/after/before. If the specified word or words are not found, the computer displays (e.g., if none of the last set of words is found) the following:

> You forgot to say when things happen in the story.
>
> You could use words like "next," "then," "after," and "before."
>
> Go back and use words that tell when things happen in your story.

Then the student text is displayed from the beginning so that the student can add specific words.

After the revision routines are completed, the computer returns to the editing routines to correct any errors that may have been made during revision. Spelling, capitalization, and punctuation are checked again (see above), and the student is directed to re-read her/his text to check for errors.

After the student has written and revised/edited her/his text, the text is printed out so that a hard copy is available.

CONCLUSION

This paper has outlined a procedure for computer-based instruction to help students to write, revise, and edit narratives. Although this outline is relatively straightforward, the computer specifications for such instruction would necessarily be rather complex (including various branchings, limitations, and termination routines). Moreover, programming for such instruction would be very sophisticated; just programming one story would take a considerable amount of time; programming a menu of stories would be a huge project. In addition, programming for a word-processor component is a major undertaking. Finally, there are hardware problems: The envisioned instruction would be most easily handled through the use of videodisc, but videodiscs currently are not in common use and are also too expensive to produce to make such instruction economically viable. Still, this paper has suggested how--given commitment of time and resources--computer-student interaction could improve writing instruction.

SAMPLE SPECIFICATIONS FOR COMPUTER INSTRUCTION IN GENERATING AND EDITING/REVISING NARRATIVE TEXT

Bruce Cronnell

n the following pages are sample specifications for computer-based instruction in generating and editing/revising narrative text. These specifications elaborate on the general specifications described in WP 2-82/02. Before actual programming could be undertaken for this instructional module, at least three tasks would need to be done:

1. The writing prompt would need to be piloted with some students at the appropriate grade level to determine what writing problems/errors the story tended to produce. Then changes would be made in "HELP" (to ensure that needed help was provided) and in "REVISION" (to ensure that practical and useful suggestions were given).
2. Word-processor/text-editor requirements would need to be specified. Then instruction would be prepared to teach students to use the word processor.
3. The nature of the student files would need to be specified.

All three tasks could be accomplished relatively easily if the need arose.

GENERATING AND EDITING/REVISING NARRATIVE TEXT

"TITLE"

1.	SWRL PRESENTS	Display for 5 seconds.
2.	WRITING A STORY	Display for 5 seconds.

If responses are found on student disk:

Go to "REVIEW."

If responses are not found on student disk:

Go to "INTRODUCTION."

"INTRODUCTION"

1.	Please type your first and last names. Be sure you type correctly. Then push the RETURN key.	Names are filed.
2.	Hi, *<first name>*. Whenever you see *<first name>*? on the screen, you need to type an answer. After you answer, always push the RETURN key. Let's practice. What grade are you in? *<first name>*?	*<first name>*: first name filed in frame 1. The grade number is filed.
3.	Good, *<first name>*. If you want to stop at any time, just push W and then push RETURN. Push RETURN.	W = whatever key(s) specified for this function.* When W and RETURN are pushed, go to "TERMINATION ROUTINES," frame 1.

Go to "WRITING."

---

*Special keys need to be designated for four functions: (1) to exit program (= W), (2) to ask for help (= X), (3) to indicate completion of writing (= Y), and (4) to scroll text (= Z). Ideally, a labeled key would be used for each function (BYE, HELP, DONE, SCROLL); however, some microcomputer keyboards do not have enough extra keys for this. Therefore, the solution may be to have the student push a function key (but not the one used for upper-case letters) and then type a word (e.g., "bye"). Details of this process must be worked out before programming is undertaken.

"REVIEW"

1\.
> Please type your first and last names. Then push the RETURN key.

If no match with names on file:

a.
> I do not know <*first & last name*>. Please type the name you used last time.

If still no match:

b.
> I still do not find the name <*first & last name*>. Please get your teacher to help me.

The computer stops, leaving the display on the screen. Teacher types in the correct name or terminates by removing the disc.

2\.
> Hello, <*first name*>.
>
> Remember:
>
> - When you see <*first name*>? on the screen, you need to type an answer.
> - After you type your answer, always push RETURN.
> - If you want to stop at any time, just push W and then push RETURN.
>
> Now push RETURN.

"REVIEW" (continued)

3\. a. After "TERMINATION ROUTINES," frame 2, or (if student was in "WRITING" or "HELP") after "TERMINATION ROUTINES," frames 1, 5, 6, 7:

Screen	Notes
Last time you were writing a story. Remember: ● If you want help, push X and then push RETURN. ● When you have finished writing your story, push Y and then push RETURN. Push RETURN.	X = whatever key(s) specified for this function; see footnote on page 2. When X and RETURN are pushed, go to "HELP." Y = whatever key(s) specified for this function; see footnote on page 2. When Y and RETURN are pushed, go to "REVISION."
Your story is about a cat and a bird. Watch to see what the cat and the bird do.	Display for 20 seconds.

Go to "WRITING," frame 2 and then to "HELP," frame 4.

b. After "TERMINATION ROUTINES," frame 3:

Screen	Notes
Last time you finished writing your story. Let's go back and revise it.	Display for 20 seconds.

Go to "REVISION."

c. After "TERMINATION ROUTINES," frame 4, or (if student was in "REVISION") after "TERMINATION ROUTINES," frames 1, 5, 6, 7:

Screen	Notes
Last time you were revising your story. Let's do some more work on your story.	Display for 20 seconds.

Go to next "REVISION" frame.

"WRITING"

1. | Today you will write a story.

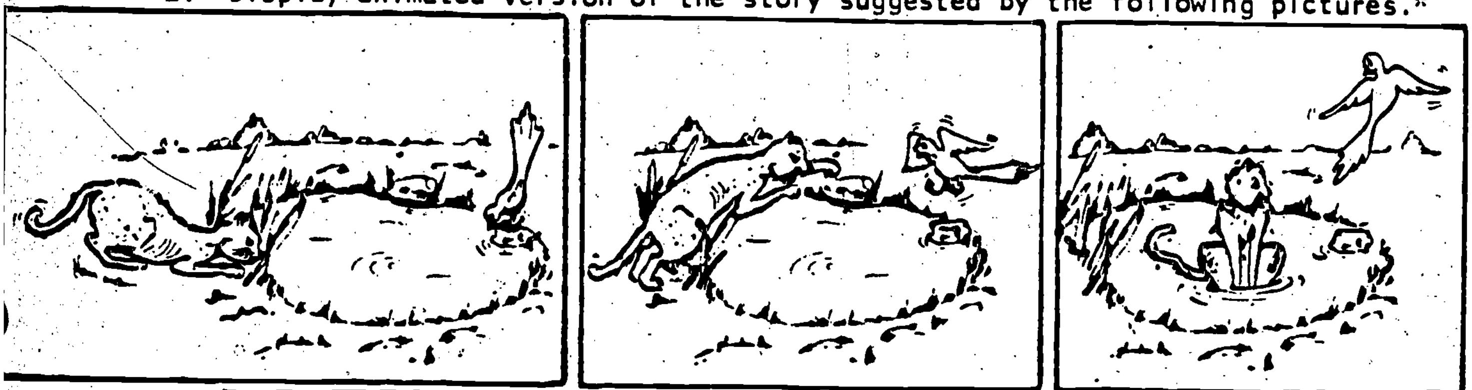
The story is about a cat and a bird.

Watch to see what the cat and the bird do.

Push RETURN. | [Picture of cat]

[Picture of bird] |
|---|---|

2. Display animated version of the story suggested by the following pictures.*



3. | Now write a story about the cat and the bird.

Tell what the cat does and and what the bird does.

Push RETURN. |
|---|

4. | If you want help, push X and then push RETURN.

When you have finished writing your story, push Y and then push RETURN.

Now begin writing.

Push RETURN. |
|---|

*Animation displayed from videotape or videodisk.

"WRITING" (continued)

Display still (three-section) version of pictures at top of screen or on separate screen. Pictures remain on screen while student is writing and whenever whole student text is displayed. (If on separate screen, pictures remain on screen except for frame 2, above.) Below pictures display title:

THE BIRD GETS AWAY

The title is the beginning of the student text; it scrolls off screen (along with text) when text fills up screen. (If pictures are on a separate screen, title also is on student screen.)

"HELP"

A. When student pushes X and RETURN, remove student text and display:

> HELP
>
> What kind of help would you like?
>
> 1. See the picture move again.
> 2. Figure out what to write next.
> 3. See some words to use in the story.
> 4. See what you have written.
> 5. Stop writing.
>
> Type the number of the kind of help you want.
>
> *<first name>*?

Go to appropriate numbered frame below.

B. When students stops typing for 60 seconds, display at bottom of screen:

> Do you want some help?
>
> Type yes or no.
>
> *<first name>*?

YES: Remove student text and go to frame A (above).

NO: Remove question (B). At bottom of screen, display for 5 seconds: WRITE SOME MORE.

1. Display "WRITING," frame 2.* Then return to student text. At bottom of screen, display for 5 seconds: NOW WRITE SOME MORE.

2. Display for 20 seconds.

> Try to think about what happens.
>
> What does the cat do?
> What does the bird do?
> Where does the story happen?
>
> Now write some more, *<first name>*.

Return to student text.

*Students may select this option several times, just for the fun of it. If this is a problem, some limitation on the use of this help feature would need to be built in.

"HELP" (continued)

3. Here are some words you can use:

cat	bird	next
water	pond	then
catch	hide	after
fall	wet	before

Now write some more, <*first name*>.

Display for 30 seconds.

Return to student text.

4\. a. If all of student text is on one screen:

Go back and read what you have written, <*first name*>.

Display for 10 seconds.

Return to student text. At bottom of screen, display for 5 seconds: NOW WRITE SOME MORE.

b. If student text is on more than one screen:

Let's go back to the beginning of your story. When you want to read more of your story, push Z.

Display for 20 seconds.

Z = whatever key(s) specified to scroll text,; see footnote on page 2.

Go to beginning of student text (i.e., from title). Scroll text as student pushes Z. Stop at end of student text and, at bottom of screen, display for 5 seconds: NOW WRITE SOME MORE.

5\. Have you finished writing your story?

Type yes or no.

<*first name*>?

YES: Go to "REVISION."

NO: Return to student text. At bottom of screen, display for 5 seconds: WRITE SOME MORE.

"REVISION"

1. When the student pushes Y and RETURN or when the student answers YES to frame 5 of "HELP":

> Now read your story again.
>
> Correct any mistakes. Make any changes that will make your story better.
>
> Push RETURN.

Display student text from beginning, with "Push Z" at bottom of each screen. When student has pushed last Z, the computer analyzes the text for each of the features in frames 2-11.

Note: Surface editing (frames 2-4) is done first--to help the computer read the text for the following frames--and then is repeated at the end for a final edit.

2. Computer has a small "dictionary" in memory (i.e., a word list of 3,000-5,000 words, including all words expected in the story).* Computer matches student text against these words (with or without capitalization and accepting forms with inflectional suffixes**). If unmatched words are found:

> You have a word that I don't understand. Please check the underlined word to be sure that it is spelled right.
>
> *<sentence>*
>
> Correct the spelling of the word and push RETURN. If the word is right, just push RETURN.

Display, one at a time, each sentence that has an unmatched word. The sentence is double spaced; the unmatched word is underlined. (If more than one unmatched word is in a sentence, the words are underlined one at a time.)

Repeat for each unmatched word.

Note: Student is allowed to continue if he/she thinks the word is right because the unmatched word may be a correct word that is not in the dictionary.

---

*There may be computer memory problems if the dictionary is included in the main program. If this is a problem, the dictionary might be a separate program that students are directed to use.

**Apparently inflectional suffixes need to be attached to words in the dictionary.

"REVISION" (continued)

3. Computer searches for capital letter at beginning of text and after each terminal punctuation (i.e., period, question mark, exclamation point). If missing:

a.

> You need a capital letter here. Please put a capital letter at the beginning of each sentence.
>
> *<text>*
>
> Push RETURN.

Computer displays text where capital letter should go: sentence needing capitalization and preceding sentence.

If student doesn't make correction within 30 seconds:

b.

> This should be a capital letter.
>
> *<text>*
>
> Push RETURN.

Same text as above, but with appropriate letter capitalized and graphically highlighted.

Repeat for each missing capital letter.

4. Computer searches for terminal punctuation before each capital letter and at end of text. (Exceptions: first word in text; "Bird," "Cat," or "Pond" preceded by "a/A" or "the/The."*) If missing:

> Do you need a punctuation mark here? Be sure you have a punctuation mark at the end of each sentence.
>
> *<text>*
>
> Push RETURN.

Computer displays text where period should go: sentence needing period and following sentence.

Repeat for each missing period.

---

*Other exceptions may need to be made here because students may misuse capital letters by placing them within sentences, and we don't want periods inserted within sentences. Moreover, students may use proper nouns correctly within sentences. I don't know a way out of this without some powerful syntax component to analyze sentences. However, to avoid inappropriate addition of periods, no automatic period placement is included in the program (as opposed to automatic capitalization in frame 3).

"REVISION" (continued)

5. Computer searches for any consecutive sequence of two sentences, both shorter than five words. (A sentence is defined as a segment beginning with capital letter and ending with period, question mark, or exclamation point, but not immediately followed by quotation marks--since short quotations are appropriate. Also excluded from this definition of sentence are the personal titles <u>Ms</u>., <u>Mr</u>., <u>Mrs</u>., and <u>Dr</u>.) If found:

These sentences are very short: *<sentences>* Can you turn them into one sentence? Fix the sentences and then push RETURN.	Computer displays both short sentences.

Repeat for each sequence of two short sentences.

6. Computer searches for any sentence longer than 20 words. If found:

This is a very long sentence: *<sentence>* Can you change this to two shorter sentences? Fix the sentence and then push RETURN.	Computer displays 20+ word sentence.

Repeat for each 20+ word sentence.

7. Computer searches for any two (or more) consecutive sentences beginning with the same word, or, in the case of sentences that begin with <u>A</u> or <u>The</u>, beginning with the same two words. If found:

> These sentences begin the same way.
>
> *<sentences>*
>
> Can you change one sentence to make the beginnings different?
>
> Fix the sentences and then push RETURN.

Computer displays sentences beginning with same word(s).

Repeat for each set of sentences beginning with the same word(s).

8. Computer searches for "bird" (with or without capitalization); if not in text:

a.

> You forgot to say that a bird is in your story.
>
> Go back and use the word "bird" in your story.
>
> Push RETURN.

Student text is displayed from beginning, with "Push 2" at bottom of each page. If "bird" does not appear within 60 seconds after the screen is displayed, display the following at the bottom of the screen (with graphic emphasis):

b.

> Please use the word "bird" in your story.

If "bird" appears or if student does not respond for 60 seconds, go on to next segment.

"REVISION" (continued)

9. Computer searches for "cat" (with or without capitalization); if not in text:

> You forgot to say that a cat is in your story.
>
> Go back and use the word "cat" in your story.
>
> Push RETURN.

Follow-up similar to that for frame 8.

10. Computer searches for "pond" and "water" (with or without capitalization); if neither is in the text:

> You forgot to say anything about the pond and the water.
>
> Go back and use the word "pond" or "water" in your story.
>
> Push RETURN.

Follow-up similar to that for frame 8.

11. Computer searches for "next," "then," "after," "before," (with or without capitalization); if none are in the text:

> You forgot to say when things happen in the story.
>
> You could use words like "next," "then," "after," and "before."
>
> Go back and use words that tell when things happen in your story.
>
> Push RETURN.

Follow-up similar to that for frame 8.

12. Repeat frames 2-4.

13. Repeat frame 1. Then go to "TERMINATION ROUTINES," frame 8.

"TERMINATION ROUTINES"

1. Student pushes W and RETURN:

> You're doing well, <*first name*>.
> You can write some more later.
>
> Goodbye, <*first name*>.

Display for 10 seconds.

Termination description:* "Session ended at student's request."

2. Student is going to "HELP" and has been on computer for more than 15 minutes:

> You're doing well, <*first name*>.
> Let's stop for today. You can write some more later.
>
> Goodbye, <*first name*>.

Display for 15 seconds.

Termination description: "Session ended at conclusion of section."

3. Student is going to "REVISION" and has been on computer for more than 15 minutes:

> You're doing well, <*first name*>.
> Let's stop for today. Next time we will revise your story.
>
> Goodbye, <*first name*>.

Display for 15 seconds.

Termination description: "Session ended at conclusion of section."

---

*Termination descriptions are placed in student's file to enable the teacher to monitor student progress.

"TERMINATION ROUTINES"

4. Student is going to a new frame in "REVISION" and has been on computer for more than 15 minutes:

> You're doing well, *<first name>*. Let's stop for today. Next time you can revise your story some more.
>
> Goodbye, *<first name>*.

Display for 15 seconds.

Termination description: "Session ended at conclusion of section."

5. Student does not respond within 15 seconds to command/question requiring a typed response:

a.

> I am waiting for your answer.

Display at bottom of screen. Sound is made when sentence is first displayed.

Student still does not respond within 30 more seconds.

b.

> I'm sorry. Goodbye.

Termination description: "Session ended because student did not respond."

6. Student does not push RETURN within 30 seconds when it must be pushed to continue:

> I'm waiting for you to push RETURN.

Display at bottom of screen. Sound is made when sentence is first displayed.

Student still does not respond within 30 more seconds:

Go to next instructional frame (i.e., act as if student had pushed RETURN).

However, if this happens twice (i.e., if student twice goes through two 30-second periods without pushing RETURN):

Display frame 5 (above).

Termination description: "Session ended because student did not push RETURN in order to continue."

"TERMINATION ROUTINES" (continued)

7. Student responds to command/question, but does not push RETURN within 15 seconds after response:

   Display frame 6 (above).

   Student still does not respond within 15 more seconds:

   Go to next instructional frame (i.e., act as if student had pushed RETURN).

   However, if this happens twice (i.e., if student twice responds and twice goes through two 15-second periods without pushing RETURN):

   Display frame 5b (above).

   Termination description: "Session ended because student did not push RETURN after making response."

<u>Note</u>: If the student is terminated by frames 5-7, he/she needs special help with using the computer. After the teacher has provided appropriate assistance, the student may go back to the computer to continue instruction.

8. Student completes "REVISION," frame 13:

Now you are finished writing and revising your story, <*first name*>. Push RETURN, and a copy of your story will be printed for you. Goodbye, <*first name*>.

   Student exits from program. Print hard copy of student story.

   Termination description: "Student has completed writing and revising a story."

PART II: ELEMENTARY COMPOSITION INSTRUCTION

PART II

SPECIFYING COMPUTER-BASED INSTRUCTION TO SUPPLEMENT ELEMENTARY SCHOOL COMPOSITION INSTRUCTION

Introduction

A. Microcomputer Instruction to Supplement a Proposed Elementary School Composition Program

B. Computer Instruction for Addressing Envelopes

1. Specifications

2. Computer Program Listing

3. Courseware

4. Pilot Study Report

PART II

SPECIFYING COMPUTER-BASED INSTRUCTION TO SUPPLEMENT ELEMENTARY SCHOOL COMPOSITION INSTRUCTION

Introduction

As part of SWRL's NIE-supported work in "Cooperative Instructional Application of Writing Research," a proposal was made for an elementary school composition program that would incorporate current research findings about the composing process and about composition instruction (see Appendix F). This Part of the report specifies how microcomputer instruction in composition might supplement such a program.

Section A outlines this supplementary instruction as a component of an elementary school composition program. Section B provides an example of such supplementary instruction. The instruction in this section uses the context of addressing envelopes to teach and practice the following skills:

Places address and return adress in the correct space on an envelope.

Capitalizes names of persons.

Capitalizes personal titles (e.g., Mr., Ms., Dr.).

Capitalizes the names of streets, cities, states.

Uses a comma between city and state.

The section includes (1) the instructional specifications (a general description, as well as a more detailed description for use in programming), (2) the computer program listing, (3) the courseware (an operating manual and a computer disk* for use with a Commodore Pet computer), and (4) a pilot study report.

*The computer disk accompanying this report has been removed.

Proposed Composition Program

MICROCOMPUTER INSTRUCTION TO SUPPLEMENT A PROPOSED ELEMENTARY SCHOOL COMPO^ ION PROGRAM

Ann Humes

MICROCOMPUTER INSTRUCTION TO SUPPLEMENT
A PROPOSED ELEMENTARY SCHOOL COMPOSITION PROGRAM

BACKGROUND AND RATIONALE

Teaching students to write is an important concern of both educators and researchers. Appropriate instruction is also a concern, for few instructional programs meet, even minimally, current classroom needs.

SWRL's proposal for elementary school composition instruction is an exception (see Appendix F). It fills the need for a unified approach to the teaching of writing. One optional element of the instructional program further addresses requirements of the contemporary classroom. This element is a set of computerized instructional materials for teaching component skills of composition. These materials can be used as supplementary practice on skills taught in composition instruction for students who do not master the skills by participating in regular classroom instruction and practice. These materials can also be used as primary instruction for some skills, particularly for students such as new enrollees who must approach these skills out of sequence with the mainline writing instruction.

The microcomputer is one of the newest and most versatile media for instruction. It can function as a teacher aide while providing the student the privacy that can make instructional tasks less threatening. It can present instruction and examples, provide interesting practice, point out information, and request clarification of ambiguities.

Microcomputer instruction offers other benefits to students and teachers. It permits truly individualized instruction because students can proceed at their own rate and can receive the appropriate amount of practice, which is determined by their computer-analyzed performance. The computer also provides immediate feedback, thus providing the student with immediate reinforcement or assistance. Additionally, microcomputer instruction is highly motivating because of its game-like nature.

STUDENT PREREQUISITES

Students must be in grades 3-6 and have previous experience with the microcomputer that will be used in the school. They must also have previous experience with typing.

CONTENT OF INSTRUCTION

Many writing skills must be employed in order to put words on a page. Writers must deal with these skills while attending to the content, structure, and coherence of their texts. Clearly, the number of skills that must be dealt with simultaneously can overload the writer's information processing capacity (Scardamalia, in Bereiter, 1979).

Research has shown that students who have mastered these skills to the point where they are relatively automatic have more freedom to focus their attention on global issues during composing (Bridwell, 1981). Furthermore, some students master these skills earlier and more quickly than other students. Consequently, individualized microcomputer instruction and practice on these skills can meet the needs of students who have not yet mastered or have not been introduced to one or more of these skills.

The skills covered here include punctuation, capitalization, language usage, and special sets of related skills. The sets of related skills are presented within products or elements of products commonly taught in elementary instructional materials. When these skills are presented in a context to which they are relevant, they can be quickly learned. The sets included are personal letters, business letters, poetry, forms such as application blanks, and dialogue for stories.

INSTRUCTIONAL COURSEWARE

The instructional courseware would be used by students in grades 3-6 and be operated on a microcomputer. The courseware would consist of a set of disks that might be used for instruction on new skills for some students and for review on previously taught skills for other students. For this reason, content is sequenced and designated as materials for grades 3-6, 4-6, and 5-6. Students in grade three would use only those materials designated for grades 3-6, which, for them would cover content first presented at grade three, with a few exceptions. Students in grade six could use any of the disks, which, for them, would have both new and review material. This design also allows for a more economical package because the same materials are not repeated on separate disks for each grade level.

For the specific content of the courseware, see the following pages.

CONTENT OF COURSEWARE FOR GRADES 3-6

PUNCTUATION

Uses a period to end a declarative sentence.
Uses a question mark to end an interrogative sentence.
Uses an exclamation point to end an exclamatory sentence.
Uses a period to end an imperative sentence.
Uses a comma in a compound sentence.
Uses commas to separate items in a series.

CAPITALIZATION

Capitalizes the first word in a sentence.
Capitalizes the personal pronoun "I."
Capitalizes the names of persons.*
Capitalizes the days of the week and holidays.
Capitalizes the first, last, and important words of a title.

USAGE

Uses regular and irregular verbs appropriately.
Uses adjectives appropriately: comparative and superlative forms.
Uses adverbs appropriately: comparative and superlative forms.
Uses articles <u>a</u> and <u>an</u> appropriately.
Uses sentences in which the subject and verb agree (subject before verb).

SPECIAL SETS OF RELATED SKILLS

<u>Personal Letters and Envelopes</u>

Uses a comma between city and state.*
Uses a comma in the date.
Punctuates the greeting and closing of a personal letter.
Capitalizes the greeting and closing of a personal letter.
Capitalizes personal titles (e.g., Mr., Ms., Dr.).*
Capitalizes the names of streets, cities, states.*
Places address and return address in the correct space on an envelope.*

COURSEWARE FOR GRADES 4-6

PUNCTUATION

Uses apostrophe in singular possessives.
Uses apostrophe in contractions.

*Taught in the specimen lessons contained in this report; see following section.

CAPITALIZATION

Capitali..es n... s of rivers, lakes, buildings.

USAGE

Uses consistent verb tense.
Uses nominative and objective pronouns appropriately.

SPECIAL SETS OF RELATED SKILLS

Forms (e.g., application blanks)

Uses periods at the end of initials.
Uses periods at the end of abbreviations.
Capitalizes initials in the names of persons.
Capitalizes names of countries.
Capitalizes abbreviations for days, months, streets.

Dialogue

Uses comma after noun of direct address.
Uses quotation marks, comma, and terminal punctuation in a dialogue quotation.
Capitalizes first word in dialogue quotations.
Uses dialogue for only one speaker in a paragraph.

Poetry

Punctuates poetry.
Capitalizes first word in line of verse.

COURSEWARE FOR GRADES 5-6

PUNCTUATION

Uses comma after introductory transitional expressions.
Uses apostrophe in plural possessive forms.
Uses a comma after introductory adverbial clauses.

CAPITALIZATION

Capitalizes first letter in nonpersonal proper names.
Capitalizes names of organizations and proper adjectives.

USAGE

Uses interrogative pronouns.
Uses past and past-participle forms of verbs appropriately.
Uses consistent verb tense.

SPECIAL SETS OF RELATED SKILLS

Business Letter

Punctuates the greeting and closing of a business letter.
Capitalizes the greeting and closing of a business letter.
Places the parts of a business letter in the correct space.
Uses appropriate formal language in a business letter.

REFERENCES

Bereiter, C. Development in writing. In R. W. Tyler & S. H. White (chairmen), Testing, teaching, and learning. Washington, D.C.: National Institute of Education, 1979, 146-166.

Bridwell, L. S. Rethinking composing. English Journal, 1981, 70:7, 96-99.

Addressing Envelopes

COMPUTER INSTRUCTION
FOR ADDRESSING ENVELOPES

1. Specifications
2. Computer Program Listing
3. Courseware
4. Pilot Study Report

SOUTHWEST REGIONAL LABORATORY
TECHNICAL NOTE

DATE: October 25, 1982

NO: TN 2-82/37

COMPUTER-BASED INSTRUCTION ON SKILLS FOR ADDRESSING ENVELOPES

Ann Humes

ABSTRACT

An approach to elementary-school computer-based instruction is presented for teaching students component placement, capitalization, and punctuation skills of addressing envelopes within the context of a simulated envelope. The skills taught are identified, and the initial orientation procedures are discussed. Then instruction and practice are outlined, and the corresponding branching for students' responses is described. A reporting system that allows the teacher to follow student progress is also described.

COMPUTER-BASED INSTRUCTION ON SKILLS FOR ADDRESSING ENVELOPES

Ann Humes

Several microcomputer programs are available that deal with math and science. However, relatively few are available for teaching students to write. Yet computers can serve a vital function in composition instruction. One approach is to use the computer to provide appropriate primary instruction and practice and appropriate supplementary practice on skills that would otherwise require considerable teacher time.

This paper sketches instruction for third and fourth grade elementary school students. The instruction teaches a set of related skills in a relevant context. The set is part of a larger design for a complete program of composition instruction that includes a set of materials for use on a microcomputer. The context of instruction is an envelope, and the skills are those needed to place the address and return address correctly on an envelope and to capitalize and punctuate the addresses.*

In the first section, this paper notes the skills taught and discusses the orientation procedures students undertake after they have loaded the program disk. It also outlines the mainline instruction and practice, describing the corresponding branching that occurs in respone to students' performance on the tasks they are presented. The next section describes the reporting system that provides information for teachers on the progress of all students using the program.

*U.S. postal abbreviations (e.g., CA for California) are not taught; zip codes are not included because of space constraints imposed by screen design that would preclude practicing capitalization for a variety of cities and states.

INSTRUCTION AND PRACTICE

The following skills are taught and reviewed:

Skills	Taught	Reviewed
Places the address and return address in the correct space on an envelope.	Lesson 1	Lesson 2
Capitalizes the names of persons	Lesson 1	Lesson 2, 3
Capitalizes personal titles (e.g., Mr.)	Lesson 1	Lesson 2, 3
Capitalizes the names of streets, cities, states.	Lesson 2	Lesson 3
Uses a comma to separate city and state.	Lesson 3	

At the beginning of each session, students type their first and last names. The computer stores the first name to use in responses to students. It stores data on students' performance under their first and last names so that (1) teachers can review students' performance and (2) the computer can place students at the proper point in the instructional sequence. If students do not respond when asked for their names, the computer requests an answer. If students still do not respond, the computer terminates the program.

When students input their names, the computer checks for the first and last names in the file. If the exact names are located, the computer places the students in Lesson 1, 2, or 3, depending on students' recorded success in the program. If students do not push the return key here or wherever this behavior is necessary to continue the program, the program reminds students to push the return key. If students still do not respond, the computer terminates the lesson.

For the first session, the computer informs students that it is waiting for their response whenever it displays their first name followed by a question mark. For subsequent sessions, the computer reminds students of this signal for input.

When a new skill is presented, the computer displays a simulated envelope with address and a return address. The computer provides instructions on the skill. After a delay for reading the rule, the computer flashes the examples of correct use to focus students' attention. When the skill is address placement, the instruction is followed by two consecutive screens with addressed envelopes, and the computer requests students to identify first the return address and then the address by typing either number one or number two. When capitalization and punctuation are taught, the instruction is followed by two consecutive displays with errors to be corrected. Throughout the program, all keys are disabled except the keys for the correct answers. Thus, students can input only correct answers.

When students do not answer correctly in the practice displays, the computer responds in one of two ways, depending on the skills. For address placement, the computer gives the correct answer and then repeats the previous instruction and presents more practice. If students do not respond correctly by the third practice display, the computer terminates the lesson. For capitalization and punctuation, the computer first tells students that they have not found all the errors, so they should keep trying. If students still do not correct all the errors, the computer repeats the previous instruction and then presents another display. If students still do not respond correctly, the computer terminates the lesson.

When students respond correctly to the practice on a skill, the computer provides positive reinforcement and then presents the next rule. When an entire lesson is successfully completed, the computer provides positive reinforcement, and promises students that they will learn more "next time," and then terminates the lesson. When students have successfully completed all lessons, the computer provides positive reinforcement and suggests the user write a letter to a friend, address an envelope, and mail the correspondence.

For skills previously taught in the program, the computer reviews the rule and presents practice. Response to correct and incorrect answers is the same as for new skills, as described above.

REPORTING SYSTEM

The reporting system provides the teacher with data on individual students and on the whole class. It also allows the teacher to delete records for individual students or for an entire class so that the disk can be reused with subsequent classes.

The teacher enters the reporting system by typing "Reports" when the computer asks for the student's name. The computer then displays a menu of options. The first option is a command for the computer to list the names of the students in the file on the disk. When this option is selected, the computer displays an alphabetized list of users' names. The teacher can request a printout of this list.

Another option listed in the reporting system menu is a command for the computer to display student reports. When this option is selected, the computer displays a student-reports menu. Selections on this menu allow the teacher (1) to review the performance of one student on the screen or on a hard copy, or (2) to obtain printed copies of all student reports. These copies can be distributed to students as "Report Cards" on their performance in addressing envelopes. An example of an individual student report is displayed in Figure 1.

```
                    ADDRESSING ENVELOPES

  BILL BURKE

  LESSON 1:  completed
    Address placement:              4 repeats
    Capitalizing titles:              perfect
    Capitalizing names:             2 repeats

  LESSON 2:  completed
    Address placement:                perfect
    Capitalizing titles
      and names:                      perfect
    Capitalizing streets,
      cities, and states:           1 repeat

  LESSON 3:  completed
    Capitalizing titles
      and names:                      perfect
    Capitalizing streets,
      cities, and states:             perfect
    Adding comma between
      city and state:                 perfect
    Review:                         2 repeats
```

Figure 1. Student Report

The third option on the Reporting System menu is a command to have the computer print out a class record. This record summarizes the performance of all students in the disk file. This form is exemplified in Figure 2.

KEY: L = Lesson; C = Complete; I = Incomplete; P = Placement; T = Titles; N = Names; St = Streets; Ci = Cities; Sa = States; Co = Commas; R = Review

CLASS RECORD

No. of Repeats*

STUDENT		L 1	P	T	N	L 2	P	T/N	St/Ci/Sa	L 3	T/N	St/Ci/Sa	Co	R
1	Joan Best	C												
2	Lisa Cook	C	4		2	C			1	C	4			2
3	John Clark	I	2											
4	Ann Drake	C	5											
5	Brad Egan	I		2										
6	Thomas Frame	C				C				I	3	2	5	2
7	Kathy Grant	I												
8	William Hall	C				C				C				
9	Carol Kane	C				C				C				
10	Stacy Salas	C			1	I	2		1					
11	Maria Sanchez	I	2											
12	Anthony Trent	C				C				C				
13	Charles Wilson	C	2			C	1			C	4			1

Figure 2. Class Record

*Repeats exceeding the number allowed for the skills in a single session indicate that the lesson was repeated.

Deleting a record is another option offered on the reporting system menu. When the teacher selects this option, the computer asks whether the teacher wants to delete records for an entire class. The teacher types "yes" or "no" in response to this question. If the teacher types "yes," the computer asks "Are you sure?" and waits for a "yes" or "no" response before records are deleted. If the teacher does not wish to delete all records, the computer asks the teacher to type the name of the student whose record is to be deleted.

The final option offered on the reporting system menu is a command to exit from the system.

This instruction has been programmed and is operating. Although it can now be used as a discrete unit of instruction, it was designed and developed as a component of a larger set of instructional materials to teach composition.

Distribution List

1 - Bailey

1 - Behr

20 - Cronnell

1 - Directorate

1 - Escoe

1 - Follettie

1 - Gentry

8 - Humes

1 - Lawlor

1 - Library

1 - Milazzo

1 - Moncrief

1 - Scott

1 - Simpkins

COMPUTER INSTRUCTION FOR ADDRESSING ENVELOPES

Ann Humes

The following are specifications of computer instruction to teach capitalization and punctuation skills within the context of a simulated envelope. These specifications elaborate on the general specifications described in TN 2-82/37. This instruction is a third-grade element of a computer-instruction component of a larger composition program. Also included are specifications of a reporting system for the instruction and the actual set of example and practice items used in the displays.

INSTRUCTION

Ann Humes

1. | ADDRESSING ENVELOPES

SWRL
Educational Research and Development
4665 Lampson Avenue
Los Alamitos, CA 90720
Copyright 1982

This material was prepared under Contract No. 400-80-0108 with the National Institute of Education, Department of Education. |

Ten-second delay before next frame.

2. | Please type your first name. Then push the RETURN key. |

3. | Now, please type your last name. |

The computer stores first name to use throughout Lesson 1. The computer stores data under student's first and last names for teacher to review.

If the student responds, but does not press return within 30 seconds, the computer branches to the termination routine. If the student does not respond within 15 seconds, the computer adds the following to the frame:

3a. | I am waiting for your answer. |

If the student does not respond within 15 seconds, the computer replaces the sentence added for 3a with the following:

3b. | If you do not answer, I must say goodbye. |

If the student does not respond within 15 seconds, the computer displays the following frame:

3c. | I am sorry. Goodbye. |

The computer terminates the lesson.

<*first name*>. In this lesson, n you see <*first name*>? on the screen, t means I am waiting for your answer. n now waiting for you to press RETURN key.	——→	If the student does not press return within 30 seconds, the computer branches to the termination routine.
envelope has two addresses. is the address the person who will get the letter. Envelope with address, but no return address s RETURN.	— — →	After a delay for reading, the address flashes several times. If the student does not press return within one minute, the computer branches to the termination routine.
other address is the return address. s the address he person who sends the letter. Envelope with both address and return address s RETURN.	— — →	After a delay for reading, return address flashes several times. If the student does not press return within one minute, the computer branches to the termination routine.

h is on the envelope? 1. address 2. return address the number for the right answer. *st name*>? Envelope with only a return address	→	The computer allows only the numeral "2" to be typed. After the correct answer is typed, the computer displays the next frame. If the student types something other than "2," the computer branches to the error routine. If the student does not type within 20 seconds, the computer adds the following to the screen: 7a. I am waiting for your answer. If the student still does not type within 20 seconds, the computer branches to the error routine.
, <*first name*>. 1 is on this envelope? 1. address 2. return address the number for the right answer. *st name*>? Envelope with only the address, no return address	→	The computer allows only the numeral "1" to be typed. After the correct answer is typed, the computer displays the next frame. If the student types something other than "1," the computer branches to the error routine. If the student does not type within 20 seconds, the computer adds the following to the screen: 8a. I am waiting for your answer. If the student still does not type within 20 seconds, the computer branches to the error routine.
good, <*first name*>. ou will learn some rules apitalizing and punctuating of addresses. RETURN.	→	If the student does not press return within 15 seconds, the computer branches to the termination routine.

ways capitalize titles
ke Dr., Mr., Ms., Mrs., and Miss.

Envelope with both addresses

ess RETURN.

Titles in rule are highlighted.
Titles on envelope flash several times after a seven-second delay for reading. If the student does not press return within 15 seconds, the computer branches to the termination routine.

the titles on this envelope.
e the cursor to the letters
t need to be capitals.
n push the correct letter.

rst name>?

Envelope with uncapitalized titles.

The computer allows only appropriate capital letters for titles to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

11a. | I am waiting for your answer. |

If the student still does not type within 30 seconds and if this is the first time through Frame 11, the computer displays the following screen:

11b. | I think you need to see the rule again. |

After a ten-second delay for reading, the computer loops back to Frame 10 with a new example.

If this is the second time through Frame 11--if the computer has looped back through Frame 10 before--the computer displays the following screen and terminates the lesson:

11c.

Let's not do this lesson now. You can work on it some other time. Goodbye, <*first name*>.

od work, <*first name*>.

ess RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

re is an important rule:
ays capitalize the names of persons.

Envelope with both address and return address

ss RETURN.

Names flash several times after a seven-second delay for reading.

If the student does not push return within 45 seconds, the computer branches to the termination routine.

e the cursor and fix the names
this envelope.

rst name>?

Envelope with names of persons uncapitalized

The computer allows only appropriate capital letters for names to be typed. After all the correct answers are typed, the computer displays the next screen. If the student types some answers correctly, but not all, the computer branches to the prompting routine after 60 seconds. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

14a. | I am waiting for your answer. |

If the student still does not type within 30 seconds and if this is the first time through Frame 14, the computer displays the following screen:

14b. | I think you need to see the rule again. |

After a delay for reading, the computer loops back to Frame 13, for a new example and then presents a new item in Frame 14.

If this is the second time through Frame 14--if the computer has looped back through Frame 13 before--the computer displays the following screen and terminates the lesson:

14c.

Let's not do this lesson now. You can work on it some other time. Goodbye, <*first name*>.

are doing well, <*first name*>. will learn more about addresses time. bye, <*first name*>.

→ The computer terminates the program.

ADDRESSING ENVELOPES

SWRL
Educational Research and Development
4665 Lampson Avenue
Los Alamitos, CA 90720
Copyright 1982

s material was prepared under Contract
400-80-0108 with the National Institute
Education, Department of Education

---> Ten-second delay before next frame.

ase type your first name.
n push the RETURN key.

, please type your last name.
push the RETURN key.

---> The computer stores first name to use throughout Lesson 2. The computer stores data under student's first and last names for teacher to review.

If the student responds, but does not press return within 30 seconds, the computer branches to the termination routine. If the student does not respond within 15 seconds, the computer adds the following to the frame:

3a. | I am waiting for your answer.

If the student does not respond within 15 seconds, the computer replaces the sentence added for 3a with the following:

3b. | If you do not answer,
I must say goodbye.

If the student does not respond with 15 seconds, the computer displays the following frame:

3c. | I am sorry.
Goodbye.

The computer terminates the lesson.

2

<first name>.
mber, when you see *<first name>*?
he screen, that means
waiting for your answer.

now waiting for you to
s the RETURN key.

If the student does not press return with 30 seconds, the computer branches to the termination routine.

mber what you have learned:

The return address is the address of the person who sends the letter.

Envelope with both address and return address

s RETURN.

After a seven-second delay for reading, the return address flashes.

If the student does not press return within 30 seconds, the computer branches to the termination routine.

other address is the address
he person who will get the letter.

Same envelope as in Frame 4 above

s RETURN.

After a seven-second delay for reading, the address flashes.

If the student does not press return within 30 seconds, the computer branches to the termination routine.

4

ich is flashing?

1. address
2. return address

pe the number of the right answer.

irst name>?

Envelope with both address and return address

The return address flashes.

The computer allows only the numeral "2" to be typed. After the correct answer is typed, the computer displays the next frame. If the student types something other than "2," the computer branches to the error routine. If the student does not type within 20 seconds, the computer adds the following to the screen:

7a. I am waiting for your answer.

If the student still does not type within 20 seconds, the computer branches to the error routine.

, <first name>.

s RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

mber that you capitalize titles Dr., Mr., Ms., Mrs., and Miss.

Envelope with return address and address

s RETURN.

Titles in rule are highlighted.
After a seven-second delay for reading, titles on envelope flash.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

er, too, that you capitalize
's names.

Same envelope as above, with
return address and address

RETURN.

After a seven-second delay for reading, names of persons flash.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

e cursor and fix the titles
people's names.

name>?

nvelope with uncapitalized
itles and names

The computer allows only appropriate capital letters for titles and names to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

11a. | I am waiting for your answer. |

If the student still does not type within 30 seconds and if this is the first time through Frame 11, the computer displays the following screen:

11b. | I think you need to see the rules again. |

After a ten-second delay for reading, the computer loops back to Frame 9 with a new example and a new item for Frame 11.

If this is the second time through the Frame 11--if the computer has looped back through Frame 9 before--the computer displays the following screen and then terminates the lesson:

11c.

Let's not do this lesson now. You can work on it some other time. Goodbye, *<first name>*.

ry good, *<first name>*. ess RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

re is an important rule: Your always capitalize the names of streets, cities, and states. Envelope with both addresses ess RETURN.

After a seven-second delay for reading, names of streets, cities, and states flash first in the return address and then in the address.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

the cursor and fix the names
e streets, cities, and states.

t *name*>?

Envelope with uncapitalized
streets, cities, states

The computer allows only appropriate capital letters for cities, streets, states to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

14a. I am waiting for your answer.

If the student still does not type within 30 seconds, and if this is the first time through Frame 14, the computer displays the following screen:

14b. I think you need to see the rule again.

After a ten-second delay for reading, the computer loops back to Frame 13 for a new example and a new item for Frame 14.

If this is the second time through the Frame 14--if the computer has looped back through Frame 13 before--the computer displays the following screen and then terminates the lesson:

14c. Let's not do this lesson now.
You can work on it some other time.
Goodbye, <*first name*>.

ood, <*first name*>.

RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

; try this one.
he capital letters.

t name>?

Envelope with streets, cities, states uncapitalized

The computer allows only appropriate capital letters for cities, streets, states to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

16a. | I am waiting for your answer. |

If the student still does not type within 30 seconds, and if this is the first time through Frame 16, the computer displays the following screen:

16b. | I think you need to see the rule again. |

After a ten-second delay for reading, the computer loops back to Frame 13 for a new example and a new item for Frame 16.

If this is the second time through the Frame 16 if the computer has looped back through Frame 13 before--the computer displays the following screen and then terminates the lesson:

16c. Let's not do this lesson now.
You can work on it some other time.
Goodbye, <*first name*>.

re doing well, <*first name*>.
ill learn more about envelopes
time.

ye, <*first name*>.

The computer terminates the lesson.

ADDRESSING ENVELOPES

SWRL
ducational Research and Development
4665 Lampson Avenue
Los Alamitos, CA 90720

material was prepared under Contract
00-80-0108 with the National Institute
ducation, Department of Education.

Ten-second delay before next frame.

e type your first name.
push the RETURN key.

please type your last name.
push the RETURN key.

The computer stores first name to use throughout Lesson 3. The computer stores data under student's first and last names for teacher to review.

If the student responds, but does not press return within 30 seconds, the computer branches to the termination routine. If the student does not respond within 15 seconds, the computer adds the following to the frame:

3a. I am waiting for your answer.

If the student does not respond within 15 seconds, the computer replaces the sentence added for 3a with the following:

3b. If you do not answer,
I must say goodbye.

If the student does not respond within 15 seconds, the computer displays the following frame:

3c. I am sorry.
Goodbye.

The computer terminates the lesson.

li, *<first name>*.
emember, when you see *<first name>*?
n the screen, that means
am waiting for your answer.

am now waiting for you to
ress the RETURN key.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

emember that you capitalize titles
ike Dr., Mr., Ms., Mrs., and Miss.

Envelope with return address and address.

ress RETURN.

Titles in rule are highlighted.

After a seven-second delay for reading, titles on envelope flash.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

emember, too, that you capitalize
eople's names.

Same envelope as above, with return address and address.

ess RETURN.

After a seven-second delay for reading, names of persons flash.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

ve the cursor and fix the titles
d the people's names.

irst name>?

Envelope with uncapitalized titles and names.

The computer allows only appropriate capital letters for titles and people's names to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen.

7a. I am waiting for your answer.

If the student still does not type within 30 seconds, and <u>if this is the first time</u> through Frame 7, the computer displays the following screen:

7b. I think you need to see the rule again.

After a delay for reading, the computer loops back to Frame 5 for a new example and a new item.

<u>If this is the second time</u> through the Frame 7--if the computer has looped back through Frame 5 before--the computer displays the following screen and then terminates the lesson:

7c.
Let's not do this lesson now.
You can work on it some other time.
Goodbye, *<first name>*.

...at's right, *<first name>*!
...ess RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

...emember that you capitalize the names
...f streets, cities, and states.

Envelope with both address and return address.

...ess RETURN.

After a seven-second delay for reading, the names of streets, cities, and states flash first in the return address and then in the address.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

fix the names
he streets, cities, and states.

st *name*>?

Envelope with names of
streets, cities, and states
uncapitalized.

The computer allows only appropriate capital letters for cities, streets, states to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

10a. I am waiting for your answer.

If the student still does not type within 30 seconds, and <u>if this is the first time</u> through Frame 10, the computer displays the following screen:

10b. I think you need to see the rule again.

After a ten-second delay for reading, the computer loops back to Frame 9 for a new example and then a new item for Frame 10.

<u>If this is the second time</u> through the Frame 10--if the computer has looped back through Frame 9 before--the computer displays the following screen and then terminates the lesson:

10c. Let's not do this lesson now.
You can work on it some other time.
Goodbye, <*first name*>.

d job, <*first name*>.
ss RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

e is the last important rule:

Use a comma
between the city and state.

Envelope with address and
return address.

ss RETURN.

After a seven-second delay for reading, the commas in both addresses flash.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

put commas where they belong.
rst name>?

Envelope with address and
return address without commas
between city and state but
space allowed for the commas.

Extra space is allowed between each word and number element in addresses. The computer allows only commas, properly placed, to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

13a. I am waiting for your answer.

good, *<first name>*.
s RETURN.

fix this envelope.
st name>?

Envelope with address and
return address without commas
between city and state but
space allowed for the commas.

If the student still does not type within 15 seconds, and if this is the first time through Frame 13, the computer displays the following screen:

13b. | I think you need to see the rule again. |

After a ten-second delay for reading, the computer loops back to Frame 12 for a new example and a new item for Frame 13.

If this is the second time through the Frame 13--if the computer has looped back through Frame 12 before--the computer displays the following screen and then terminates the lesson:

13c.

> Let's not do this lesson now.
> You can work on it some other time.
> Goodbye, *<first name>*.

If the student does not respond within 15 seconds, the computer branches to the termination routine.

The computer allows only commas, properly placed, to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 45 seconds, the student types some answers correctly, but not all, the computer branches to the prompting routine. If the student does not enter one correction within 30 seconds, the computer adds the following to the screen:

15a. | I am waiting for your answer. |

If the student still does not type within 30 seconds, and if this is the first time through Frame 15, the computer displays the following screen:

15b. | I think you need to see the rule again. |

After a ten-second delay for reading, the computer loops back to Frame 12 for a new example and new items for Frame 13 and for Frame 15.

If this is the second time through Frame 15, the computer displays the following screen and then terminates the lesson:

15c. | Let's not do this lesson now. You can work on it some other time. Goodbye, <*first name*>. |

work, <*first name*>.
RETURN.

If the student does not press return within 15 seconds, the computer branches to the termination routine.

ave learned so much
envelopes.
fix
the mistakes in this one
st name>?

Envelope with errors in
punctuation, and capitalization
for all the rules.

The computer allows only correct changes to be typed. After all the correct answers are typed, the computer displays the next screen. If, within 20 seconds, the student types some answers correctly, but not all, the computer flashes and corrects the errors, one at a time. Then the computer displays the following screen:

17a. | You have learned much about addressing envelopes, <*first name*>. I think you will learn more next time. Goodbye, <*first name*>. |

The computer terminates the lesson.

was super, *<first name>*!
ou know how to address envelopes.
no: write a letter to a friend.
address an envelope and mail the
r.
fun, *<first name>*.

If the student does not start typing within 30 seconds, the computer adds the following to the screen:

17b. | I am waiting for your answer. |

If the student still does not type within 30 seconds, the computer displays the following screen and then terminates the lesson:

17c. | Let's not do any more now.
You can work on the lesson some other time.
Goodbye, *<first name>*. |

The computer terminates the lesson.

TERMINATION ROUTINE

se push RETURN.
ou do not push RETURN,
st say goodbye.

If the student pushes RETURN, the computer branches back to the next frame in the instructional sequence.

If the student does not push RETURN, the computer displays the following:

I am sorry. Goodbye.

The computer then terminates the lesson.

ERROR ROUTINE

correct answer is <   >.

need to study some more.

The computer displays the correct answer. After a delay, the computer branches to Frame 5 in the instructional sequence, <u>if this is the first or second time</u> through the error routine. <u>If this is the third time</u> through the error routine, the computer displays the following frame:

> Let's do this lesson some other time.
>
> Goodbye, <*first name*>.

The computer then terminates the lesson.

have not found all the mistakes.
se keep trying.

et name>?

This display is added to bottom of screen displayed before branching.

If the student corrects all the errors, the computer branches back to display the next screen in instructional sequence.

If the student does not correct all the errors, the computer adds the following sentence:

Here is/are the other mistake(s).

The computer flashes errors and corrects them one at a time. Then the computer displays the following frame:

You may need to study more.
Look at the rule again.

The computer branches to the instructional frame or frames immediately preceding the item frame that caused this branching to the prompting routines:

If branching is caused by →	Branch back to
Lesson 1, Frame 11	Frame 10
Lesson 1, Frame 14	Frame 13
Lesson 2, Frame 11	Frame 9
Lesson 2, Frame 14	Frame 13
Lesson 2, Frame 16	Frame 13
Lesson 3, Frame 7	Frame 5
Lesson 3, Frame 10	Frame 9
Lesson 3, Frame 13	Frame 12
Lesson 3, Frame 15	Frame 12

REPORTING SYSTEM

Ann Humes

Reporting System Menu

lease type your first name.
hen push the Return Key.

When this screen appears at first of instruction, teacher types in "Reports."

MENU	
ist students in file.	Push 1
isplay Student Reports.	Push 2
rint Class Record.	Push 3
elete a record.	Push 4
xit Reporting System.	Push 5

When 1 is pushed, student list is displayed. When 2 is pushed, computer branches to Student Reports. When 3 is pushed, computer prints the class record. When 4 is pushed, computer branches to Delete System. When 5 is pushed, computer terminates system and displays cursor for input calling back instruction or another program.

Student Reports

To see the report for any student, type the name that is recorded in the Student List and push Return.

For copies of reports for all students in file, push P and then Return.

To return to menu for Reporting System, push R and then Return.

Please type the student's first name, and then push the Return Key.

When P is pushed, copies of all reports are printed. When R is pushed, computer branches to Reporting System menu. If user inputs a name, the computer displays the following screen:

1a. Now, please type the student's last name and push the Return Key.

The computer then locates the student file and displays the Student Report. At the bottom of the report, the computer displays the following:

1b. Push P for a copy of this report. Push R to return for another entry.

When P is pushed, a copy of report is printed. When R is pushed, computer displays Student Reports menu.

Do you want to [illegible] all student record[illegible]
T[illegible] yes or no and push Return.
Push R and then Return to see the menu.

If R is typed, computer branches to Reporting System menu. If "yes" is typed, computer displays the following frame:

1a. Are you sure? Type yes or no. Then push Return.

If "yes" is again typed, the computer, before returning to Reporting System menu, displays following frame to verify the deletion.

1b. <Class records deleted.>

If "no" is typed for frame 1 or 1a, the computer displays the following frame:

1c. Please type the student's first name and then push the Return Key.
Push R and then Return to see the menu.

If R is typed, the computer branches to the Reporting System menu. If a name is typed, the computer then displays the following frame:

1d. Now, please type the student's last name and push the Return Key.
Push R and then Return to see the menu.

If the name is found, the computer deletes the corresponding record and displays this frame before returning to the menu for student reports:

Student record deleted.

If the name is not found, the computer displays the following frame before returning to the menu for student reports:

This name is not in the file.

Form for Class Record Sheet

KEY: L = Lesson; C = Complete; I = Incomplete; P = Placement; T = Titles; N = Names; St = Streets; Ci = Cities; Sa = States; Co = Commas; R = Review

CLASS RECORD

No. of Repeats

STUDENT	L 1	P	T	N	L 2	P	T/N	St/Ci/Sa	L 3	T/N	St/Ci/Sa	Co	R
<name, name>													

SAMPLE STUDENT REPORT

ADDRESSING ENVELOPES

BILL BURKE

LESSON 1: completed	
Address placement:	4 repeats
Capitalizing titles:	perfect
Capitalizing names:	2 repeats
	2 repeats
LESSON 2: completed	
Address placement:	perfect
Capitalizing titles and names:	perfect
Capitalizing streets, cities and states:	1 repeat
LESSON 3: completed	
Capitalizing titles and names:	perfect
Capitalizing streets, cities, and states:	perfect
Adding comma between city and state:	perfect
Review	2 repeats

Examples and Items

for

Addressing Envelopes

Larry Gentry

LESSON 1

Frame		Examples	Frame		Items
4	(to)	Mary Brown 135 Main Street Canton, Ohio	6	(from)	Betty Gomez 857 First Avenue Price, Idaho
5	(from)	Bob Jones 244 Oak Street Bend, Oregon	7	(to)	Gary Martin 702 Keel Circle Downey, Texas
	(to)	Mary Brown (see above)			
9 Ex. 1	(from)	Dr. Jan Marks 842 Dale Avenue Sparks, Nevada	10 Item 1	(from)	mr. Tom Smith 362 Spring Street Dover, Vermont
	(to)	Mr. Dan Shaw 976 Olson Drive Ogden, Utah		(to)	ms. Karen Mack 1537 Lewis Road Harper, Kansas
9 Ex. 2	(from)	Miss Sue Baker 259 Park Street Puna, Hawaii	10 Item 2	(from)	mrs. Nancy Ray 715 Third Street Casper, Oregon
	(to)	Mr. Frank Quinn 440 Harbor Avenue Bayport, Maine		(to)	dr. Jose Luna 1312 Elm Drive Reed, Montana
12 Ex. 1	(from)	Mr. Jim Reese 175 Palm Drive Kern, Texas	13 Item 1	(from)	Ms. irene white 820 Titus Way Hope, Arizona
	(to)	Mrs. Becky Wong 307 Trask Road Nome, Alaska		(to)	Mr. john short 403 Wilson Road Morton, Utah
12 Ex. 2	(from)	Ms. Alice Neal 296 Fourth Street Stuart, Iowa	13 Item 2	(from)	Mr. paul west 111 Maple Street Anchor, Maine
	(to)	Mr. Ben Scott 831 Beach Street Milton, Ohio		(to)	Miss carol morris 1853 Arden Way Post, Nevada

LESSON 2

Frame		Examples
4	(from)	Mr. Carlos Garcia 555 Pine Avenue Tibbs, Iowa
	(to)	Mrs. Judy Kaplan 1066 Olson Drive Boise, Idaho
5		(same as above)
8 Ex. 1	(from)	Ms. Ruth Dana 369 Ocean Avenue Dalton, Maine
	(to)	Dr. Roy Marsh 1654 Main Street Weston, Maine
Ex. 1 9		(same as above)
8 Ex. 2	(from)	Mr. Ken Tanaka 935 Plaza Lane Yuma, Arizona
	(to)	Mrs. Edna Hatch 268 Sixth Avenue Miami, Florida
Ex. 2 9		(same as above)
12 Ex. 1	(from)	Mr. Dennis Small 503 Adams Street Dallas, Texas
	(to)	Miss Ellen Lopez 219 Fern Avenue Preston, Ohio
12 Ex. 2	(from)	Ms. Kathy Evans 942 Olive Way Butte, Montana
	(to)	Mr. Bill Johnson 730 Circle Drive Westport, Alaska

Frame		Items
6	(from)	Mrs. Julie Mann 792 Fifth Street Page, Utah
	(to)	Mr. David Black 416 Fern Road Midway, Kansas
10 Item 1	(from)	dr. ann mason 2034 Olive Avenue Colton, Alabama
	(to)	mr. james ward 480 Cherry Drive Paris, Indiana
10 Item 2	(from)	mrs. donna rice 638 Fulton Street Jackson, Ohio
	(to)	ms. maria vega 9810 Central Avenue Monroe, Georgia
13 Item 1	(from)	Mrs. Joan Irwin 302 south street reno, nevada
	(to)	Mr. Henry Chang 2295 bell avenue hilo, hawaii
13 Item 2	(from)	Mr Greg Holt 934 cadet drive fenway, vermont
	(to)	Dr. Debbie Starr 668 hill street bixby, kansas

LESSON 2 (continued)

Frame		Examples	Frame		Items
12 Ex. 3	(from)	Mr. George Kent 1313 Ellis Road Welton, Utah	15 Item 1	(from)	Miss Sandy Adams 581 temple avenue troy, alabama
	(to)	Mr. Dale Pierce 100 Willow Street Newton, Iowa		(to)	Mr. Mark Welch 1133 first street baker, oregon
			15 Item 2	(from)	Mrs. Millie Dean 387 summit way nome, alaska
				(to)	Dr. Jean Thomas 592 dalton avenue willis, iowa

LESSON 3

Frame			Examples
4&5	Ex. 1	(from)	Mr. Ray Knox 222 Campus Road Waco, Texas
		(to)	Dr. Doris Stein 1234 Main Street Dallas, Texas
4&5	Ex. 2	(from)	Ms. Gloria Cole 7456 Larson Drive Dayton, Ohio
		(to)	Mrs. Jane Beal 307 Berry Circle Tampa, Florida
8	Ex. 1	(from)	Mr. Glen Romero 649 Harvard Street Midland, Iowa
		(to)	Miss Peggy Duke 143 Banner Avenue Kona, Hawaii
8	Ex. 2	(from)	Ms. Ana Duran 9275 Lilac Road Dover, Maine
		(to)	Mr. Walter Olson 583 Crosby Lane Galway, Utah
11	Ex. 1	(from)	Mr. Alex Little 889 Dudley Way Weston, Idaho
		(to)	Mrs. Lisa Roe 705 Adams Street Plains, Georgia
11	Ex. 2	(from)	Mr. Andy Briggs 1355 Pearl Way Newton, Alabama
		(to)	Mr. Tony Ricco 432 Poplar Lane Walnut, Nevada

Frame			Items
6	Item 1	(from)	dr. howard king 2801 Center Street Gary, Indiana
		(to)	mr. steve jacobs 806 Sherman Way Murray, Idaho
6	Item 2	(from)	mr. fred shiner 476 Benton Avenue Jackson, Alaska
		(to)	ms. helen gray 8463 Simon Drive Clinton, Kansas
9	Item 1	(from)	Miss Amy Jarvis 592 madison road grange, oregon
		(to)	Mr. Aaron Mills 6482 dewey drive atlanta, georgia
9	Item 2	(from)	Mr. Wayne Tucker 375 dilday way carson, montana
		(to)	Mrs. Dina Burns 462 forest lane lubbock, texas

LESSON 3 (continued)

Frame		Examples
11 Ex. 3	(from)	Ms. Lucy Brand 541 Ketch Street Jasper, Idaho
	(to)	Dr. Eric Long 603 Royal Avenue Ludlow, Texas

Frame		Items
12 Item 3	(from)	Mr. Roger Wood 625 Vera Lane Dolan Oregon
	(to)	Mrs. Margie Hale 1060 Pyle Circle Teal Indiana
12 Item 4	(from)	Miss Kim Ming 864 Munson Road Bangor Maine
	(to)	Mr. Robert Keene 936 Preston Street Barrow Alaska
14 Item 1	(from)	Dr. Wes Harvey 782 Hope Drive Fenway Ohio
	(to)	Ms. Esther Lopez 433 Flower Street Jerome Kansas
14 Item 2	(from)	Miss Sharon Laird 856 Grand Avenue Hana Hawaii
	(to)	Mr. Albert Dixon 150 Jade Street Manley Iowa
16	(from)	mr. brian davis 643 crystal avenue elko nevada
	(to)	dr. june carter 1837 ludlow street logan utah

COMPUTER PROGRAM FOR ENVELOPE-ADDRESSING SKILLS

Jerry Bailey

On the following pages is the listing for the computer program for envelope-addressing skills. It is a 24k BASIC program written to run on a Commodore microcomputer with 32k of memory, a graphics keyboard, and a disk drive. The code is very linear and follows the instructional specifications (see preceding section) frame for frame except for the address-correction routine. This routine was placed at the front of the program to maximize its speed.

The correction routine is the only part of the program with machine-specific code. The pokes, peeks, and the values assigned to x, which are used to control the cursor, may need to be changed if this program is used with a Commodore 64 or one of the Commodores with a business keyboard.

One of the features of this program is that the shift key need never be used by the student. This feature dictated that the normal cursor-left key could not be used because it is a shift of the cursor-right key. The left and right angle brackets are convenient substitutes when using the graphics keyboard.

The disk contains both a commented and an uncommented version of the program. The uncommented version is used when running the program because it loads slightly more quickly and runs faster. The commented version is for information purposes only.

The "pupil file" generated by this program and used by the reports program has the following record format:

BYTES	CONTENTS
1-15	Last name
16-30	First name
31	Last lesson used (1-3)
32	Completion code for lesson 1 (c = complete; 1-5 = last part attempted)
33-35	Repeats on placement
36-38	Repeats on titles
39-41	Repeats on names
42-47	not used
48	Completion code for lesson 2
49-51	Repeats on placement
52-54	Repeats on titles and names
55-57	Repeats on streets/cities/states
58-63	not used
64	Completion code for lesson 3

BYTES	CONTENTS
65-67	Repeats on titles and names
68-70	Repeats on streets/cities/states
71-73	Repeats on commas
74-76	Repeats on review
77-79	not used

ENVELOPE ADDRESSING

```
1 open15,8,15:goto100
2 rem
3 rem  "Display item and count things to be corrected
4 pr=0:c=0:cc=0:gosub58000:fori=1to6:j=i>3
5 printleft$(v$,tt+i-2*j)spc(lz+1-10*j)d$(i):r(i)=len(d$(i))
6 iflen(c$(i))=0thennexti:goto12
7 forj=1tolen(c$(i)):ifmid$(c$(i),j,1)<>o$thencc=cc+1
8 nextj,i
10 rem
11 rem    "Item correction routine
12 r=1:p=1:k=1:t3=ti+1800:tm=ti+4800:ifcc<12thentm=ti+3600:ifcc<3thentm=ti+2700
13 j=r>3:printleft$(v$,tt+r-2*j)spc(lz+p-10*j)
14 poke168,1:poke167,0:ifpeek(151)=xthenforq=0tou:next:u=0:ifpeek(151)=xgoto19
15 geta$:x=0:u=199:ifc=0theniftiؚ>t3goto42
16 ifctheniftiؚ>tmgoto51000
17 ifa$=""goto15
18 rem  "Turn off cursor and rewrite character to eliminate reverse field
19 poke167,1:printmid$(d$(r),p,1)"[illegible]";
20 ifa$="<"ora$=" "goto26
21 ifa$=">"ora$=" "goto30
22 if(asc(a$)and127)=(asc(mid$(c$(r),p,1)+n$)and127)goto34
23 poke167,0:goto15:rem  turn cursor back on
24 rem
25 rem  "Move cursor left
26 x=5:p=p-1:ifp<1thenr=r-1:p=r(r):ifr<1thenr=6:p=r(6)
27 goto13
28 rem
29 rem  "Move cursor right
30 x=12:p=p+1:ifp>r(r)thenr=r+1:p=1:ifr>6thenr=1
31 goto13
32 rem
33 rem  "Count correction--fix error--set so can't fix again
34 c=c+1:printmid$(c$(r),p,1)"[illegible]";:
35 d$(r)=mid$(left$(o$+d$(r),p)+mid$(c$(r),p,1)+mid$(d$(r),p+1),2)
36 c$(r)=mid$(left$(o$+c$(r),p)+o$+mid$(c$(r),p+1),2):ifc<ccgoto13
37 rem "Corrected all errors so flash boarder
38 vv=23:gosub61400:print"[illegible]That's right, "n$".
39 forj=1to5:gosub58200:gosub61500:gosub58100:gosub61500:next:return
40 rem
41 rem  "Prompt for slow response.
42 poke167,1:printmid$(d$(r),p,1)"[illegible]";
43 ifkthenprintleft$(v$,24)"I am waiting for your answer.":k=0:t3=ti+1800:goto1
3
44 ifit>1goto57
45 printleft$(v$,24)"I think you need to see the rule again.":gosub57000:te=5:r
eturn
55 rem
56 rem  "Bail out if 2nd time through
57 print"[illegible]Let's not do this lesson now.
58 print"[illegible]You can work on it some other time.
59 print"[illegible]Goodbye, "n$".":te=4:goto62090
90 rem
92 rem
94 rem "Line numbers mod 2000 and then divided by 100 match frame numbers
96 rem
98 rem "Display SWRL screen
100 poke59468,14:v$="[illegible]":ad=17:ra=7:o$=">":lz=6:tt=10
102 sl=0:dimd$(6),c$(6),r(6):cl$="
                                                                        [illegible]"
```


BEST COPY AVAILABLE

```
104 print"█"spc(10)"ADDRESSING ENVELOPES
106 printspc(16)"█":rem  concatesson 1
108 printspc(18)"█SWRL
110 printspc(2)"Educational Research and Development
112 printspc(10)"4665 Lampson Avenue
114 printspc(9)"Los Alamitos, CA 90720
116 printspc(13)"Copyright 1982
118 printspc(3)"███This material was prepared under
120 printspc(3)"Contract No. 400-80-0108 with the
122 printspc(3)"National Institute of Education,
124 printspc(3)"Department of Education.
126 ti$="000000":tw=600+ti:ifaagoto140
128 rem
130 rem  "Read pupil file while waiting
132 open8,8,8,"0:pupil file"
134 dim p$(100),p(5)
136 n=n+1:input#8,p$(n):ifst=0goto136
138 close8:iflen(p$(1))=0thenn=0
140 gosub57004:n$=chr$(0):i=1:j=1:k=1:m=1:de=900:te=0
196 rem
198 rem        "Input pupil names
200 print"██Please type your first name":gosub50000:iftegoto62050
202 nf$=s$:i=1:j=1:k=1:m=1:ifs$="REPORTS"thenload"0:report prog",8
204 vv=5:gosub61400:print"████"nf$
206 print"██Now, please type your last name":gosub50000:iftegoto62050
208 nl$=s$
210 rem
212 rem  "Sort name into list and determine proper lesson
214 na$=left$(nl$+cl$,15)+left$(nf$+cl$,15):ifn=0thenpp=1:goto222
216 fori=1ton:ifna$>p$(i)thennext
218 pp=i:ifna$=left$(p$(i),30)theni=n:next:goto223
220 fori=ntoppstep-1:p$(i+1)=p$(i):next
222 n=n+1:le=1:p$(pp)=na$+"1":goto300
223 le=val(mid$(p$(pp),31,1)):ls$=mid$(p$(pp),16*(1+le),1):ls=val(ls$)
224 ifls$="c"thenle=le+1:goto226
225 fori=1to5:p(i)=val(mid$(p$(pp),15+le*15+i*3,3)):next
226 onlegoto300,2300,2300
230 print:print"████You have completed all three lessons,█":printnf$"."
232 goto62135
296 rem
298 rem  "Explain name prompt
300 print"█Hi, "nf$".  In this lesson,
302 print"When you see "nf$"? on the
304 print"screen, that means I am waiting":print"for your answer.█"
306 print"I am now waiting for you to press the":print"RETURN key.█":printnf$"?
308 de=1800:gosub61900
396 rem
398 rem  "Display envelop with adressee only and explain
400 sl=1:gosub52000
401 print"█An envelope has two addresses.":print"One is the address
402 print"of the person who will get the letter."
403 j=1:td=450:gosub58000:goto405
404 gosub58000:gosub61500
405 gosub61800:gosub57002:gosub61210:ifj<6goto404
406 de=3600:gosub61900
496 rem
498 rem  "Display with both addresses and explain return address
500 print"█The other address is the return address.";
502 print"It is the address":print"of the person who sends the letter.
504 gosub58000:gosub61800:j=0:td=450:goto510
```

BEST COPY AVAILABLE

```
506 gosub580:gosub61500
510 printleft$(v$,11)spc(ra)"Bob Jones":printspc(ra)"244 Oak Street
512 printspc(ra)"Bend, Oregon":gosub57002:gosub61210:ifj<6goto506
514 gosub61900:de=900:iftegoto62050
516 goto600
580 printleft$(v$,11)spc(ra)"                    "
582 printspc(ra)"                    ":printspc(ra)"                    ":return
596 rem
598 rem  "Quiz on return address
600 print"":gosub58000
602 printleft$(v$,11)spc(ra)"Betty Gomez":printspc(ra)"857 First Avenue"
604 printspc(ra)"Price, Idaho"
606 print"Which is on the envelope?"
608 an$="2":gosub61600:iferthengosub52010:goto401
696 rem
698 rem  "Quiz on addressee
700 print"":gosub58000
702 printleft$(v$,16)spc(ad)"Gary Martin":printspc(ad)"702 Keel Circle
704 printspc(ad)"Downey, Texas"
706 print"Good, "nf$".":print"Which is on this envelope?"
708 an$="1":gosub61600:iferthengosub52010:goto401
796 rem
798 rem  "Reinforce and say what will do next
800 print"Very good, "nf$"."
802 print"Now you will learn some rules
804 print"for capitalizing and punctuating
806 print"parts of addresses."
808 gosub61900
896 rem
898 rem  "Capitalize titles
900 sl=2:gosub52000:gosub948:gosub58000
902 j=1:td=450:goto908
904 printleft$(v$,11)spc(ra)"   "
906 printleft$(v$,16)spc(ad)"   ":gosub61500
908 printleft$(v$,11)spc(ra)"Dr. Jan Marks":printspc(ra)"842 Dale Avenue
910 printspc(ra)"Sparks, Nevada
912 printleft$(v$,16)spc(ad)"Mr. Dan Shaw":printspc(ad)"976 Olson Drive
914 printspc(ad)"Ogden, Utah":gosub57002
916 gosub61210:ifj<6goto904
918 gosub61900
920 it=1:goto1000
922 rem
924 rem  2nd example if fail quiz
926 gosub948
928 gosub58000
930 j=1:td=450:goto936
932 printleft$(v$,11)spc(ra)"   "
934 printleft$(v$,16)spc(ad)"   ":gosub61500
936 printleft$(v$,11)spc(ra)"Miss Sue Baker":printspc(ra)"259 Park Street
937 printspc(ra)"Puna, Hawaii
938 printleft$(v$,16)spc(ad)"Mr. Frank Quinn":printspc(ad)"440 Harbor Avenue
940 printspc(ad)"Bayport, Maine":gosub57002
942 gosub61210:ifj<6goto932
944 gosub61900
946 it=2:goto1000
948 print"Always capitalize titles
950 print"like Dr., Mr., Ms., Mrs., and Miss.
952 print"     -     -     -     -               -":return
996 rem
998 rem  "Capitalize titles quiz item
```

BEST COPY AVAILABLE

```
1000 print"■Fix the titles on this envelope.
1002 print"Move the cursor to the letters
1004 print"that need to be capitals.
1006 print"Then push the correct letter.■
1008 printn$"?":gosub53000:onitgoto1010,1048
1010 d$(1)="mr. Tom Smith
1012 c$(1)="M
1014 d$(2)="362 Spring Street
1018 d$(3)="Dover, Vermont
1022 d$(4)="ms. Karen Mack
1024 c$(4)="M
1026 d$(5)="1537 Lewis Road
1030 d$(6)="Harper, Kansas
1038 te=0:gosub4:iftethengosub58010:goto926
1040 goto1100
1044 rem
1046 rem  2nd item if fail first one
1048 d$(1)="mrs. Nancy Ray
1050 c$(1)="M
1052 d$(2)="715 Third Street
1056 d$(3)="Casper, Oregon
1060 d$(4)="dr. Jose Luna
1062 c$(4)="D
1064 d$(5)="1312 Elm Drive
1068 d$(6)="Reed, Montana
1072 te=0:gosub4
1096 rem
1098 rem   "Reinforce
1100 sl=3:gosub52000:print"■Good work, "n$".■":gosub61900
1196 rem
1198 rem  "Capitalize names rule
1200 gosub1254:gosub58000
1202 j=0:td=450:goto1212
1204 print left$(u$,11)spc(ra)"Mr.            "
1208 print left$(u$,16)spc(ad)"Mrs.           ":gosub61500
1212 print left$(u$,11)spc(ra)"Mr. Jim Reese":printspc(ra)"175 Palm Drive
1214 printspc(ra)"Kern, Texas
1216 print left$(u$,16)spc(ad)"Mrs. Becky Wong":printspc(ad)"307 Trask Road
1218 printspc(ad)"Nome, Alaska":gosub61210:gosub57002
1220 ifj<6goto1204
1222 de=2700:gosub61900
1224 it=1:goto1300
1226 rem
1228 rem  2nd example if fail first quiz
1230 gosub1254:gosub58000
1232 j=1:td=450:goto1238
1234 print left$(u$,11)spc(ra)"Ms.            "
1236 print left$(u$,16)spc(ad)"Mr.            ":gosub61500
1238 print left$(u$,11)spc(ra)"Ms. Alice Neal":printspc(ra)"296 Fourth Street
1240 printspc(ra)"Stuart, Iowa
1242 print left$(u$,16)spc(ad)"Mr. Ben Scott":printspc(ad)"831 Beach Street
1244 printspc(ad)"Milton, Ohio":gosub57002:gosub61210
1246 ifj<6goto1234
1248 gosub61900
1250 it=2:goto1300
1252 rem
1254 print"■Here is an important rule:
1256 print"Always capitalize the names of persons.":return
1296 rem
1298 rem  "Capitalize names quiz item
```

```
1300 print"Move the cursor and fix the names
1302 print"on this envelope.":printnf$"?":gosub53000:onitgoto1304,1334
1304 d$(1)="Ms. irene white
1306 c$(1)=">>>>I>>>>>W
1308 d$(2)="820 Titus Way
1312 d$(3)="Hope, Arizona
1316 d$(4)="Mr. john short
1318 c$(4)=">>>>J>>>>S
1320 d$(5)="403 Wilson Road
1324 d$(6)="Morton, Utah
1328 te=0:gosub4:iftethengosub52010:goto1230
1330 goto1400
1334 d$(1)="Mr. paul west
1336 c$(1)=">>>>P>>>>W
1338 d$(2)="111 Maple Street
1342 d$(3)="Anchor, Maine
1346 d$(4)="Miss carol morris
1348 c$(4)=">>>>>C>>>>>M
1350 d$(5)="1853 Arden Way
1354 d$(6)="Post, Nevada
1358 te=0:gosub4
1396 rem
1398 rem   "OK end message
1400 print"You are doing very well, "nf$"."
1402 print"You will learn more about addresses
1404 print"next time.
1406 print"Goodbye, "nf$".":goto62090
2198 rem  "Explain name prompt
2300 print"Hi, "nf$".":print"Remember, when you see "nf$"?"
2302 print"on the screen, that means
2304 print"I am waiting for your answer."
2306 print"I am now waiting for you to press the":print"RETURN key.":printnf$"
?
2308 de=1800:gosub61900:ifle>2goto4400
2396 rem
2398 rem  "Display envelop with both addresses and give reminder
2400 sl=1:gosub52000
2401 print"Remember what you have learned:
2402 print"  The return address is the address
2404 print"  of the person who sends the letter.
2406 gosub58000:j=1
2408 printleft$(v$,16)spc(ad)"Mrs. Judy Kaplan
2410 printspc(ad)"1066 Olson Drive":printspc(ad)"Boise, Idaho":td=450:goto2420
2412 printleft$(v$,16)spc(ad)"Mrs. Judy Kaplan
2416 printleft$(v$,11)spc(ra)"                    "
2418 printspc(ra)"                    ":printspc(ra)"                    ":gosub61500
2420 printleft$(v$,11)spc(ra)"Mr. Carlos Garcia":printspc(ra)"555 Pine Avenue
2422 printspc(ra)"Tibbs, Iowa":gosub57002:gosub61210:ifj<6goto2416
2424 gosub61900:de=900:ifteyoto62050
2496 rem
2498 rem   "Remind about addressee
2500 print"The other address is the address
2502 print"of the person who will get the letter.
2506 gosub58000:j=1
2512 printleft$(v$,11)spc(ra)"Mr. Carlos Garcia":printspc(ra)"555 Pine Avenue
2514 printspc(ra)"Tibbs, Iowa":td=450:goto2520
2516 printleft$(v$,16)spc(ad)"                    "
2518 printspc(ad)"                    ":printspc(ad)"                    ":gosub61500
2520 printleft$(v$,16)spc(ad)"Mrs. Judy Kaplan
2522 printspc(ad)"1066 Olson Drive":printspc(ad)"Boise, Idaho"
```

```
2524 gosub57002:j=j+1:gosub61210:ifj<6goto2516
2526 gosub61900:de=900:iftegoto62050
2596 rem
2598 rem    "Quiz on return address
2600 print"":gosub58000:gosub2680:printleft$(v$,16)spc(ad)"Mr. David Black
2610 printspc(ad)"416 Fern Road":printspc(ad)"Midway, Kansas
2620 an$="2":print"Which is flashing?":gosub61600:iferthengosub52010:goto2401
2622 goto2700
2680 printleft$(v$,11)spc(ra)"Mrs. Julie Mann"
2682 printspc(ra)"792 Fifth Street":printspc(ra)"Page, Utah":return
2696 rem
2698 rem    "Reinforce correct response
2700 print"Good "nf$".":gosub61900:it=0:sl=2:gosub52000
2796 rem
2798 rem    "Reminder to capitalize titles
2800 gosub2826:j=1:td=450:it=it+1:onitgoto2816,2846
2804 printleft$(v$,11)spc(ra)"    "
2806 printleft$(v$,16)spc(ad)"    ":gosub61500
2816 gosub2870:gosub61210:ifj<6goto2804
2818 gosub61900
2820 goto2900
2826 print"Remember that you capitalize titles":gosub950:goto58000
2834 printleft$(v$,11)spc(ra)"    "
2836 printleft$(v$,16)spc(ad)"    ":gosub61500
2846 gosub2880:gosub61210:ifj<6goto2834
2848 gosub61900
2850 goto2900
2870 printleft$(v$,11)spc(ra)"Ms. Ruth Dana":printspc(ra)"369 Ocean Avenue
2872 printspc(ra)"Dalton, Maine
2874 printleft$(v$,16)spc(ad)"Dr. Roy Marsh":printspc(ad)"1654 Main Street
2876 printspc(ad)"Weston, Maine":gosub57002:return
2880 printleft$(v$,11)spc(ra)"Mr. Ken Tanaka":printspc(ra)"935 Plaza Lane
2882 printspc(ra)"Yuma, Arizona
2884 printleft$(v$,16)spc(ad)"Mrs. Edna Hatch":printspc(ad)"268 Sixth Avenue
2886 printspc(ad)"Miami, Florida":gosub57002:return
2896 rem
2898 rem    "Reminder to capitalize names
2900 gosub2926:j=1:td=450:onitgoto2916,2946
2904 printleft$(v$,11)spc(ra)"Ms.            "
2906 printleft$(v$,16)spc(ad)"Dr.            ":gosub61500
2916 gosub2870:gosub61210:ifj<6goto2904
2918 gosub61900
2920 goto3000
2926 print"Remember, too, that you capitalize"
2928 print"people's names.":goto58000
2934 printleft$(v$,11)spc(ra)"Mr.            "
2936 printleft$(v$,16)spc(ad)"Mrs.           ":gosub61500
2946 gosub2880:gosub61210:ifj<6goto2934
2948 gosub61900
2950 goto3000
2996 rem
2998 rem    "Titles and names item
3000 gosub3040:onitgoto3010,3050
3010 d$(1)="dr. ann mason"
3012 c$(1)="D>>>A>>>M
3014 d$(2)="2034 Olive Avenue
3018 d$(3)="Colton, Alabama
3022 d$(4)="mr. james ward
3024 c$(4)="M>>>J>>>>>W
3026 d$(5)="400 Cherry Drive
```

```
3030 d$(6)="Paris, Indiana
3034 te=0:gosub4:iftethengosub52010:goto2800
3036 goto3100
3040 print"Move the cursor and fix the titles":print"and the people's names."
3042 printnf$"?":goto53800
3046 rem
3048 rem    2nd item if fail first one
3050 d$(1)="mrs. donna rice
3052 c$(1)="M>>>>D>>>>>R
3054 d$(2)="638 Fulton Street
3058 d$(3)="Jackson, Ohio
3062 d$(4)="ms. maria vega
3064 c$(4)="M>>M>>>>>V
3066 d$(5)="9910 Central Avenue
3070 d$(6)="Monroe, Georgia
3074 te=0:gosub4
3096 rem
3098 rem    "Reinforce
3100 sl=3:gosub52000:print"Very good, "nf$".":gosub61900:it=0
3196 rem
3198 rem    "More capitalization
3200 print"Here is an important rule:"
3202 print"  You always capitalize the names
3204 print"  of streets, cities, and states.
3206 gosub58000:j=1:td=450:it=it+1:onitgoto3212,3232,3252
3208 printleft$(v$,12)spc(ra)"503                    "
3210 printspc(ra)"                           ":gosub61500
3212 gosub3270:gosub61210:ifj<6goto3208
3214 j=1
3216 printleft$(v$,17)spc(ad)"219                    ":printspc(ad)"
"
3218 gosub61500
3220 gosub3274:gosub61210:ifj<6goto3216
3222 gosub61900:goto3300
3228 printleft$(v$,12)spc(ra)"942                    "
3230 printspc(ra)"                           ":gosub61500
3232 gosub3280:gosub61210:ifj<6goto3228
3234 j=1
3236 printleft$(v$,17)spc(ad)"730                    ":printspc(ad)"
"
3238 gosub61500:gosub3284:gosub61210:ifj<6goto3236
3240 gosub61900:goto3300
3248 printleft$(v$,12)spc(ra)"1313                   "
3250 printspc(ra)"                           ":gosub61500
3252 gosub3290:gosub61210:ifj<6goto3248
3254 j=1
3256 printleft$(v$,17)spc(ad)"100                    ":printspc(ad)"
"
3258 gosub61500:gosub3292:gosub61210:ifj<6goto3256
3260 gosub61900:it=2:goto3500
3270 printleft$(v$,11)spc(ra)"Mr. Dennis Small":printspc(ra)"503 Adams Street
3272 printspc(ra)"Dallas, Texas
3274 printleft$(v$,16)spc(ad)"Miss Ellen Lopez":printspc(ad)"219 Fern Avenue
3276 printspc(ad)"Preston, Ohio":goto57002
3280 printleft$(v$,11)spc(ra)"Ms. Kathy Evans":printspc(ra)"942 Olive Way
3282 printspc(ra)"Butte, Montana
3284 printleft$(v$,16)spc(ad)"Mr. Bill Johnson":printspc(ad)"730 Circle Drive
3286 printspc(ad)"Westport, Alaska":goto57002
3290 printleft$(v$,11)spc(ra)"Mr. George Kent":printspc(ra)"1313 Ellis Road
3291 printspc(ra)"Welton, Utah
```

```
3292 printleft$(v$,16)spc(ad)"Mr. Dale Pierce":printspc(ad)"100 Willow Street
3293 printspc(ad)"Newton, Iowa":goto57002
3296 rem
3298 rem    "Streets, cities, and states capitalization quiz
3300 print"Move the cursor and fix the names
3302 print"of the streets, cities, and states.":printnf$"?":gosub58000
3304 gosub53000:onitgoto3310,3350
3310 d$(1)="Mrs. Joan Irwin
3314 d$(2)="302 south street
3316 c$(2)=">>>>S>>>>>S
3318 d$(3)="reno, nevada
3320 c$(3)="R>>>>>N
3322 d$(4)="Mr. Henry Chang
3326 d$(5)="2295 bell avenue
3328 c$(5)=">>>>>B>>>>A
3330 d$(6)="hilo, hawaii
3332 c$(6)="H>>>>>H
3334 te=0:gosub4:iftethengosub52010:goto3200
3342 goto3400
3346 rem
3348 rem    2nd item if fail first one
3350 d$(1)="Mr. Greg Holt
3354 d$(2)="934 cadet drive
3356 c$(2)=">>>>C>>>>>D
3358 d$(3)="fenway, vermont
3360 c$(3)="F>>>>>>>V
3362 d$(4)="Dr. Debbie Starr
3366 d$(5)="668 hill street
3368 c$(5)=">>>>H>>>>S
3370 d$(6)="bixby, kansas
3372 c$(6)="B>>>>>>K
3374 te=0:gosub4
3396 rem
3398 rem    "Reinforce
3400 print"Very good, "nf$".":gosub61900:it=1
3496 rem
3498 rem    "Streets, cities, and states capitalization 2nd quiz
3500 print"Let's try this one.":print"Fix the letters that
3502 print"should be capitals.":gosub53000:onitgoto3510,3550
3510 d$(1)="Miss Sandy Adams
3514 d$(2)="581 temple avenue
3516 c$(2)=">>>>T>>>>>>A
3518 d$(3)="troy, alabama
3520 c$(3)="T>>>>>A
3522 d$(4)="Mr. Mark Welch
3526 d$(5)="1133 first street
3528 c$(5)=">>>>>F>>>>>S
3530 d$(6)="baker, oregon
3532 c$(6)="B>>>>>>O
3534 te=0:gosub4:iftethengosub52010:it=2:goto3200
3542 goto3600
3546 rem
3548 rem    2nd item if fail first one
3550 d$(1)="Mrs. Millie Dean
3554 d$(2)="307 summit way
3556 c$(2)=">>>>S>>>>>>W
3558 d$(3)="nome, alaska
3560 c$(3)="N>>>>>A
3562 d$(4)="Dr. Jean Thomas
3566 d$(5)="592 dalton avenue
```

```
3568 c$(5)=">>>>D>>>>>>A
3570 d$(6)="willis, iowa
3572 c$(6)="W>>>>>>>I
3574 te=0:gosub4:goto3600
3596 rem
3598 rem    "Did it right now terminate using lesson 1 code
3600 goto4400
4396 rem
4398 rem  "Capitalize titles reminder
4400 sl=1:gosub52000:it=0
4410 gosub2826:j=1:td=450:it=it+1:onitgoto4418,4438
4414 gosub4450
4418 gosub4470:gosub57002:gosub61210:ifj<6goto4414
4420 gosub61900:goto4500
4434 gosub4450
4438 gosub4480:gosub57002:gosub61210:ifj<6goto4434
4440 goto4420
4450 printleft$(v$,11)spc(ra)"     ":printleft$(v$,16)spc(ad)"     ":goto61500
4470 printleft$(v$,11)spc(ra)"Mr. Ray Knox":printspc(ra)"222 Campus Road
4472 printspc(ra)"Waco, Texas
4474 printleft$(v$,16)spc(ad)"Dr. Doris Stein":printspc(ad)"1234 Main Street
4476 printspc(ad)"Dallas, Texas":return
4480 printleft$(v$,11)spc(ra)"Ms. Gloria Cole":printspc(ra)"7456 Larson Drive
4482 printspc(ra)"Dayton, Ohio
4484 printleft$(v$,16)spc(ad)"Mrs. Jane Beal":printspc(ad)"307 Berry Circle
4486 printspc(ad)"Tampa, Florida":return
4496 rem
4498 rem  "Capitalize names reminder
4500 gosub2926:j=1:td=450:onitgoto4518,4538
4514 printleft$(v$,11)spc(ra)"Mr.               "
4516 printleft$(v$,16)spc(ad)"Dr.               ":gosub61500
4518 gosub4470:gosub57002:gosub61210:ifj<6goto4514
4520 gosub61900:goto4600
4534 printleft$(v$,11)spc(ra)"Ms.               "
4536 printleft$(v$,16)spc(ad)"Mrs.              ":gosub61500
4538 gosub4480:gosub57002:gosub61210:ifj<6goto4534
4540 goto4520
4596 rem
4598 rem  "Capitalize names and titles
4600 gosub3040:onitgoto4610,4650
4610 d$(1)="dr. howard king
4612 c$(1)="D>>>H>>>>>>K
4614 d$(2)="2001 Center Street
4618 d$(3)="Gary, Indiana
4622 d$(4)="mr. steve jacobs
4624 c$(4)="M>>>S>>>>>J
4626 d$(5)="806 Sherman Way
4630 d$(6)="Murray, Idaho
4634 te=0:gosub4:iftethengosub52010:goto4410
4636 goto4700
4650 d$(1)="mr. fred shiner
4652 c$(1)="M>>>F>>>>S
4654 d$(2)="476 Benton Avenue
4658 d$(3)="Jackson, Alaska
4662 d$(4)="ms. helen gray
4664 c$(4)="M>>>H>>>>>G
4666 d$(5)="8463 Simon Drive
4670 d$(6)="Clinton, Kansas
4674 te=0:gosub4
4696 rem
```

```
4698 rem    "Reinforce
4700 g1=2:gosub52000:print"That's right, "n$".":gosub61900:it=0
4796 rem
4798 rem    "Capitalize steets, cities, and states reminder
4800 print"Remember that you capitalize the names
4802 print"of streets, cities, and states.
4805 gosub58000:j=1:td=450:it=it+1:onitgoto4812,4832
4808 printleft$(v$,12)spc(ra)"649                    "
4810 printspc(ra)"                    ":gosub61500
4812 gosub4870:gosub61210:ifj<6goto4808
4814 j=1
4816 printleft$(v$,17)spc(ad)"143                    "
4818 printspc(ad)"                    "
4820 gosub61500:gosub4874:gosub61210:ifj<6goto4816
4822 gosub61900:goto4900
4828 printleft$(v$,12)spc(ra)"9275                   "
4830 printspc(ra)"                    ":gosub61500
4832 gosub4880:gosub61210:ifj<6goto4830
4834 j=1
4836 printleft$(v$,17)spc(ad)"583                    "
4838 printspc(ad)"                    "
4840 gosub61500:gosub4884:gosub61210:ifj<6goto4836
4842 goto4822
4870 printleft$(v$,11)spc(ra)"Mr. Glen Romero":printspc(ra)"649 Harvard Street
4872 printspc(ra)"Midland, Iowa
4874 printleft$(v$,16)spc(ad)"Miss Peggy Duke":printspc(ad)"143 Banner Avenue
4876 printspc(ad)"Kona, Hawaii":goto57002
4880 printleft$(v$,11)spc(ra)"Ms. Ana Duran":printspc(ra)"9275 Lilac Road
4882 printspc(ra)"Dover, Maine
4884 printleft$(v$,16)spc(ad)"Mr. Walter Olson":printspc(ad)"583 Crosby Lane
4886 printspc(ad)"Galway, Utah":goto57002
4896 rem
4898 rem    "Capitalize steets, cities, and states quiz
4900 print"Now fix the names
4902 print"of the streets, cities, and states.":printn$"?"
4904 gosub53000:onitgoto4910,4950
4910 d$(1)="Miss Amy Jarvis
4914 d$(2)="592 madison road
4916 c$(2)=">>>>M>>>>>>>R
4918 d$(3)="grange, oregon
4920 c$(3)="G>>>>>>>O
4922 d$(4)="Mr. Aaron Mills
4926 d$(5)="6482 dewey drive
4928 c$(5)=">>>>>D>>>>>D
4930 d$(6)="atlanta, georgia
4932 c$(6)="A>>>>>>>>G
4934 te=0:gosub4:iftethengosub52010:goto4800
4936 goto5000
4950 d$(1)="Mr. Wayne Tucker
4954 d$(2)="375 gilday way
4956 c$(2)=">>>>D>>>>>>W
4958 d$(3)="carson, montana
4960 c$(3)="C>>>>>>>M
4962 d$(4)="Mrs. Dina Byrns
4966 d$(5)="462 forest lane
4968 c$(5)=">>>>F>>>>>>L
4970 d$(6)="lubbock, texas
4972 c$(6)="L>>>>>>>>T
4974 te=0:gosub4
4996 rem
```

```
4998 rem     "Reinforce
5000 sl=3:gosub52000:print"■Good job, "nf$".":gosub61900:it=0
5096 rem
5098 rem     "Comma between city and state rule
5100 print"■Here is the last important rule:■":print"  Use a comma
5102 print"  between the city and state.
5104 gosub50000:j=1:td=450:it=it+1:onitgoto5110,5130,5150
5106 forj=1to5:cx$=" ":fori=1to2:gosub57002:printleft$(v$,13)spc(ra+cr)cx$
5108 td=45:printleft$(v$,18)spc(ad+ca)cx$:cx$=",":nexti,j:return
5110 printleft$(v$,11)spc(ra)"Mr. Alex Little":printspc(ra)"889 Dudley Way
5112 printspc(ra)"Weston, Idaho
5114 printleft$(v$,16)spc(ad)"Mrs. Lisa Roe":printspc(ad)"705 Adams Street
5116 printspc(ad)"Plains, Georgia":cr=6:ca=6:gosub5106:gosub61900:goto5200
5130 printleft$(v$,11)spc(ra)"Mr. Andy Briggs":printspc(ra)"1335 Pearl Way
5132 printspc(ra)"Newton, Alabama
5134 printleft$(v$,16)spc(ad)"Mr. Tony Ricco":printspc(ad)"432 Poplar Lane
5136 printspc(ad)"Walnut, Nevada":cr=6:ca=6:gosub5106:gosub61900:goto5200
5150 printleft$(v$,11)spc(ra)"Ms. Lucy Brand":printspc(ra)"541 Ketch Street
5152 printspc(ra)"Jasper, Idaho
5154 printleft$(v$,16)spc(ad)"Dr. Eric Long":printspc(ad)"603 Royal Avenue
5156 printspc(ad)"Ludlow, Texas":cr=6:ca=6:gosub5106:gosub61900:it=2:goto5400
5196 rem
5198 rem     "Comma quiz
5200 print"■Now put commas where they belong.■":printnf$"?"
5204 gosub53000:onitgoto5210,5250
5210 d$(1)="Mr. Roger Wood
5214 d$(2)="625 Vera Lane
5218 d$(3)="Dolan  Oregon
5220 c$(3)=">>>>>,
5222 d$(4)="Mrs. Margie Hale
5226 d$(5)="1060 Pyle Circle
5230 d$(6)="Teal  Indiana
5232 c$(6)=">>>>,
5234 te=0:gosub4:iftethengosub52010:goto5100
5236 goto5300
5250 d$(1)="Miss Kim Ming
5254 d$(2)="864 Munson Road
5258 d$(3)="Bangor  Maine
5260 c$(3)=">>>>>>,
5262 d$(4)="Mr. Robert Keene
5266 d$(5)="936 Preston Street
5270 d$(6)="Barrow  Alaska
5272 c$(6)=">>>>>>,
5274 te=0:gosub4
5296 rem
5298 rem     "Reinforce
5300 print"■Very good, "nf$".":gosub61900:it=1
5396 rem
5398 rem     "More comma quiz
5400 print"■Now fix this envelope.■":printnf$"?"
5404 gosub53000:onitgoto5410,5450
5410 d$(1)="Dr. Wes Harvey
5414 d$(2)="782 Hope Drive
5418 d$(3)="Fenway  Ohio
5420 c$(3)=">>>>>>,
5422 d$(4)="Ms. Esther Lopez
5426 d$(5)="433 Flower Street
5430 d$(6)="Jerome  Kansas
5432 c$(6)=">>>>>>,
5434 te=0:gosub4:iftethengosub52010:it=2:goto5100
```

```
5436 soto5500
5450 d$(1)="Miss Sharon Laird
5454 d$(2)="856 Grand Avenue
5458 d$(3)="Hana  Hawaii
5460 c$(3)=">>>>,
5462 d$(4)="Mr. Albert Dixon
5466 d$(5)="150 Jade Street
5470 d$(6)="Manley  Iowa
5472 c$(6)=">>>>>>,
5474 te=0:sosub4
5496 rem
5498 rem    "Reinforce
5500 print"■Nice work, "nf$".":sl=4:sosub52000:sosub61900
5596 rem
5598 rem    "Whole ball of wax
5600 print"■You have learned so much":print"about envelopes.  Now fix
5602 print"all the mistakes in this one.■":printnf$"?"
5610 d$(1)="mr. brian davis
5612 c$(1)="M>>>B>>>>>D
5614 d$(2)="643 crystal avenue
5616 c$(2)=">>>>C>>>>>>>A
5618 d$(3)="elko  nevada
5620 c$(3)="E>>>,>N
5622 d$(4)="dr. june carter
5624 c$(4)="D>>>J>>>>C
5626 d$(5)="1837 ludlow street
5628 c$(5)=">>>>>L>>>>>>S
5630 d$(6)="logan  utah
5632 c$(6)="L>>>>,>U
5634 te=0:it=-9:sosub4:ifte=0soto5700
5640 print"■You have learned much about addressins■
5642 print"envelopes, "nf$".■
5644 print"I think you will learn more next time■■":soto59
5696 rem
5698 rem    "Finale
5700 print"■That was super, "nf$"!■
5702 print"Now you know how to address envelopes.■
5704 print"Why not write a letter to a friend.■
5706 print"Then address an envelope and mail the■
5707 print"letter.  Have your teacher help you■
5708 print"with the zip code.■
5710 print"Have fun, "nf$".":soto62090
49990 rem
49992 rem
49994 rem    "SUBROUTINES
49996 rem
49998 rem    "Name input--timins--prompts for delayed response
50000 print"and then press the RETURN key.■
50005 s$="":l=0:ll=0:t=ti+de
50010 setа$:s=asc(a$+n$):ifs>63ands<91thens=s+128
50012 ifll<>lthenll=l:t=ti+de:rem  reset timer for every key lesal key
50014 ifithenifjandti>tthent=ti+de:print"I am waiting for your answer.■":j=0
50016 ifithenifkandti>tsoto50080
50018 ifithenifmandti>tthente=1:return
50019 ifll=landti>t+10thensoto62000
50020 ifs>192ands<219andl<15thena$=chr$(s):s$=s$+a$:l=l+1:i=0:printa$;:soto5001
0
50030 ifland(s=13ors=141)thenreturn
50040 ifs=148ors=20thenifthenprint"■";:s$=left$(s$,l-1):l=l-1
50050 soto50010
```

```
50080 print"If you do not answer,                    ":print"I must say goodbye."
50090 t=t+de:k=0:goto50010
50996 rem
50998 rem   "Item slow response prompt
51000 poke167,1:printmid$(d$(r),p,1)"";:ifprgoto51500
51005 forj=1to3:gosub61500:vv=23:gosub61400:gosub61500
51010 printleft$(v$,23)"You have not found all the mistakes."
51020 print"Please keep trying.":printcl$
51030 printnf$"?":next:pr=1:tm=ti+2700:goto13adv.
51496 rem
51498 rem   "Too long on item so show rest of errors
51500 vv=23:gosub61400:forj=1to3:gosub61500:printleft$(v$,24)cl$:gosub61500
51503 ifcc-c>1thenprint"Here are the other mistakes.            ":goto51510
51506 print"Here is the other mistake.              "
51510 next:fori=1to6:j=i>3:iflen(c$(i))=0thennexti:goto51700
51520 fork=1tolen(c$(i)):c$=mid$(c$(i),k,1):ifc$=o$thennextk,i:goto51700
51530 printleft$(v$,tt+i-2*j)spc(k+lz-10*j);
51540 forii=1to10:print" ";:forjj=1to50:next:printc$"";:forjj=1to50
51550 nextjj,ii,k,i
51700 te=6:gosub57000:ifit>1goto57:rem  exit through bail out message on 2nd ti
me
51705 ifit<0thenreturn:rem  special exit for ball of max item
51710 print"You may need to study more.
51720 print"Look at the rule again.":te=8:gosub57000:return
51722 rem    return goes back to calling frame with error flag set
51996 rem
51998 rem   "Error counter
52000 ifls<>slthenreturn
52010 [illegible])+1:return:print"counting repeat":gosub57000:return
52996 rem
52998 rem   "Null correction array
53000 fori=1to6:c$(i)="":return
56996 rem
56998 rem   "Delay -- 10 seconds nominal -- shifted space will abort
57000 td=600
57002 tm=ti+td
57004 geta$:ifa$=" "thenj=9e9:return
57006 ifti<tmgoto57004
57010 return
57996 rem
57998 rem   "Envelope boarder--also used to clear interior of envelope
58000 printleft$(v$,10)"          ┌────────────────────────────────┐
58010 fori=1to11:print"      ]                                ]":next
58020 print"      └────────────────────────────────┘":return
58100 printleft$(v$,10)"          ┌────────────────────────────────┐
58110 fori=1to11:print"       ]██████████████████████████████]":next
58120 print"      └────────────────────────────────┘":return
58200 printleft$(v$,10)"     **********************************
58210 fori=1to11:print"     *██████████████████████████████*":next
58220 print"     **********************************":return
58996 rem
58998 rem   "Clear keyboard buffer
59000 forzz=0to9:geta$:next:return
61196 rem
61198 rem   "Test for shifted space.
61200 geta$
61210 j=j+1:ifa$=" "thenj=9e9
61220 return
61396 rem
61398 rem   "Clear bottom three lines
```

```
61400 forvv=vvto25:printleft$(v$,vv)cl$:next:return
61496 rem
61498 rem  .75 second delay for flashing
61500 td=45:goto57002
61596 rem
61598 rem  "Type of address quiz display and input processing
61600 er=0:oo=2:tf=ti+45:print"
61610 print"    1.  address
61620 print"    2.  return address
61630 print"Type the number for the right answer.":printnf$"?":i=0:t=ti+1200
61640 geta$:ifchr$(asc(a$+n$)and127)=an$thenreturn
61645 ifle=2andti>tfthenonoogosub2680,580:oo=3-oo:tf=ti+45
61650 ifti<tgoto61680
61660 ifithengosub61700:er=2:return
61670 printleft$(v$,24)"I am waiting for your answer.":i=1:t=ti+1200:goto61640
61680 ifa$=""goto61640
61690 i=1:goto61660
61700 printleft$(v$,23)"The correct answer is "an$".
61710 print"You may need to study some more.":te=0:gosub57000
61730 ex=ex+1:ifex<3thenreturn
61740 print"Let's do this lesson some other time."
61750 print"Goodbye, "nf$".":te=5:goto62090
61796 rem
61798 rem  "Addressee display
61800 printleft$(v$,16)spc(ad)"Mary Brown":printspc(ad)"135 Main Street
61810 printspc(ad)"Canton, Ohio":return
61896 rem
61898 rem  "RETURN prompt at bottom of screen and timing processing
61900 printleft$(v$,25)"Press RETURN.";:gosub62015:iftethengosub62000:iftegoto6
2050
61910 return
61996 rem
61998 rem  "Final RETURN prompt and processing
62000 printleft$(v$,23)"Please push RETURN.":print"If you do not push RETURN,
62010 print"I must say goodbye.";
62015 te=0:t=ti+de:gosub59000
62020 geta$:ifa$=chr$(13)ora$=chr$(141)thenreturn
62030 ifti<tgoto62020
62040 te=2:return
62050 print"I am sorry.  Goodbye.
62086 rem
62088 rem  "Termination routine
62090 ifpp=0thengosub57000:qq=1:goto104:rem loop if no pupil
62092 ifsl=0thensl=ls:rem      set last sublesson if have used lesson before
62095 a$=left$(p$(pp),16*le+15):c$="c":iftethenc$=right$(str$(sl),1)
62097 a$=left$(a$,30)+right$(str$(le),1)+mid$(a$,32)+c$
62100 fori=1to5:a$=a$+right$("  "+str$(p(i)),3):next
62110 p$(pp)=a$:open8,8,8,"@0:pupil file,s,w"
62120 fori=1topn:print#8,p$(i)chr$(13);:input#15,en,em$:ifen<19goto62130
62122 print"disk error  "em$
62124 print"Pupil file improperly updated."
62126 print"Call the teacher to correct the error."
62128 print"Then press the shift key to finish.":close8:wait152,1:goto62110
62130 nexti:close8
62135 gosub57000:run
ready.
```

REPORT PROGRAM

```
4 close15:open15,8,15
5 open4,4:open3,3:poke59468,14
10 print"     ENVELOPE ADDRESSING PUPIL REPORT"
15 printspc(17)"MENU"
30 print" List students in file.         Push 1
40 print" Display Student Reports.       Push 2
50 print" Print Class Record.            Push 3
60 print" Delete student or class.       Push 4
70 print" Exit Reporting System.         Push 5
80 print"(reading student data--please wait)
90 z$=chr$(0):b$="                          ":v$="                          "
100 ifnsoto130:rem if called from addressing program skip read in
105 dimp$(100):open8,8,8,"0:pupil file"
110 n=n+1:input#8,p$(n):ifst=0goto110
120 close8:iflen(p$(1))<30thenn=-1
130 print""b$b$
150 f=3
160 rem f=4:d$=""+chr$(1)
200 geta$:a=asc(a$+chr$(48))and127-48:onagoto5000,220,6000,8000,9000:goto200
220 print""spc(10)"STUDENT REPORTS"
221 print"To see the report for any student,
222 print"type the name that is recorded in the
223 print"Student List and push Return."
224 print"For copies of reports for all students
225 print"in file, push P and then Return."
226 print"To return to menu for Reporting System, push R and then Return.
227 print"Please type the student's first name":gosub50000:n$=left$(s$+b$,15)
228 ifs$="R"goto10
229 ifs$="P"thenf=4:d$=""+chr$(1):forp=1ton:gosub480:next:goto220
230 print"Now, type the student's last name":gosub50000:n$=left$(s$+b$,15)+n$
240 forj=1ton:ifn$>left$(p$(j),30)thennext:goto270
250 p=j:j=n:next:ifn$=left$(p$(p),30)thenf=3:d$="":gosub500:goto220
270 printleft$(v$,24)"This name is not in the file.
280 print"Press RETURN to enter another name."
290 geta$:ifa$=chr$(13)ora$=chr$(141)goto220
295 goto290
400 forp=1ton:gosub500:next:goto220
480 print"Printing reports for every student.":goto510
500 print"";
510 iff=4thenforj=1to3:print#f:next:print#f,b$d$"ADDRESSING ENVELOPES":print#f
515 gosub1500:rem name
520 l=1:gosub2000:rem lesson line
540 print#f,d$"  Address placement:               ";:s=1:gosub3000:rem sublesson
550 print#f,d$"  Capitalizing titles:             ";:s=2:gosub3000
560 print#f,d$"  Capitalizing names:              ";:s=3:gosub3000
570 ifcthenj=6:goto800
600 l=2:print#f
620 gosub2000:rem lesson line
640 print#f,d$"  Address placement:               ";:s=1:gosub3000:rem sublesson
650 print#f,d$"  Capitalizing titles
655 print#f,d$"     and names:                    ";:s=2:gosub3000
660 print#f,d$"  Capitalizing streets,            "
665 print#f,d$"     cities, and states            ";:s=3:gosub3000
670 ifcthenj=13:goto800
700 l=3:print#f
720 gosub2000:rem lesson line
750 print#f,d$"  Capitalizing titles
755 print#f,d$"     and names:                    ";:s=1:gosub3000
```

```
760 print#f,d$"  Capitalizing streets,         "
765 print#f,d$"    cities, and states:         ";:s=2:gosub3000
770 print#f,d$"  Adding comma between          "
775 print#f,d$"    city and state:             ";:s=3:gosub3000
780 print#f,d$"  Review:                       ";:s=4:gosub3000:j=22
800 iff=4thenfork=jto27:print#f:next:return
805 printleft$(v$,24)"  Push P for a copy of this report.
810 print"   Push R to return for another entry."
820 geta$:ifa$="p"ora$="P"thenf=4:d$=""+chr$(1):goto510
830 ifa$<>"r"anda$<>"R"thengoto820
950 return
1500 n$=left$(p$(p),15):gosub1580:n1$=n$
1510 n$=mid$(p$(p),16,15):gosub1580:n$=n$+" "+n1$
1520 print#f,d$n$:print#f:return
1580 ifright$(n$,1)=" "thenn$=left$(n$,len(n$)-1):goto1580
1590 return
2000 print#f,d$"LESSON"str$(l)":  ";
2005 cc$=mid$(p$(p),16+16*l,1):cc=val(cc$):ifcc$<>"c"thenprint#f,"not ";
2010 print#f,"completed":ifcc=0thencc=99
2020 c=val(mid$(p$(p),31,1))=l:return:rem c flags last lesson
3000 ifcc<sthenprint#f:return
3005 r$=mid$(p$(p),14+16*l+3*s,3):rem number of repeats
3010 ifr$="  0"andcc=sgoto3040
3011 ifr$="  0"thenprint#f,"  perfect";:goto3040
3020 print#f,r$" repeat";
3030 ifr$<>"  1"thenprint#f,"s";
3040 print#f:return
4000 f=4:forp=1ton
4010 print#f,""right$("  "+str$(p),3)"  "mid$(p$(p),16,15)left$(p$(p),15)
4015 ifint(p/60)=p/60thenforj=1to6print#f:nextj:rem make page break
4020 next:f=3:goto10
5000 f=3:p=1
5005 print"";:j=p+20:ifj>nthenj=n+1
5010 print#f,right$("  "+str$(p),3)"  "mid$(p$(p),16,15)left$(p$(p),15)
5015 p=p+1:ifp<jgoto5010
5020 ifp<=ngoto5100
5030 print"Push P for a copy of this list.
5035 print"Push R to return to the menu."
5040 geta$:ifa$="p"ora$="P"goto4000
5050 ifa$<>"r"anda$<>"R"goto5040
5060 goto10
5100 print"Push RETURN to see more names.
5110 geta$:ifa$<>chr$(13)goto5110
5120 goto5005
6000 print"      Printing a class report."
6001 f=4:forp=1ton:p$=p$(p):s$=" ":goto6023
6002 print#f,""right$("  "+str$(p),3)"  "mid$(p$,16,9)" "left$(p$,15);
6004 print#f,s$mid$(p$,31,1);:j=32:gosub6900
6006 print#f,mid$(p$,33,9);:j=48:gosub6900
6010 print#f,mid$(p$,49,12);:j=64:gosub6900
6012 print#f,mid$(p$,65,12)
6015 ifint(p/60)=p/60thenforj=1to6print#f:nextj:rem make page break
6020 next:f=3:goto10
6023 z=(p-1)/50:ifz<>int(z)goto6050
6024 ifp>1thenforj=1to9:print#f:next
6025 print#f,"KEY:  L = Lesson; C = Complete; 1 = Incomplete;";
6026 print#f," P = Placement; T = Titles;
6027 print#f,"       N = Names; St = Streets; Sa = States; Co = Commas;";
6028 print#f," R = Review":print#f:print#f:print#f,left$(b$+b$,35)"CLASS RECOR
D
```

```
6029 print#f:print#f:print#f,left$(b$+b$,34)"No. of Repeats":print#f
6030 print#f,left$(b$+b$,31)"L"left$(b$,11)"L"left$(b$,17)"L"
6035 print#f,"     STUDENT";
6040 print#f,left$(b$+b$,19)"1  P  T  H  2 P T/N St/Ci/Sa  3 T/N St/Ci/Sa Co R
6045 print#f
6050 print#f,""right$("   "+str$(p),3)"  "mid$(p$,16,9)" "left$(p$,15)s$;
6060 j=32:gosub6900
6070 j=33:lk=32:k=35:gosub6800
6080 k=38:gosub6800
6090 k=41:gosub6800:print#f,s$s$;
6100 j=48:gosub6900
6110 j=49:lk=44:k=46:gosub6800:ifj=0goto6190
6120 k=49:gosub6800
6130 k=56:gosub6800:print#f,left$(b$,5);
6140 j=64:gosub6900
6150 j=65:lk=62:k=65:gosub6800:ifj=0goto6190
6160 k=72:gosub6800
6170 k=78:gosub6800
6180 k=80:gosub6800
6190 print#f:nextp:goto10
6800 l=k-lk-1:z=val(mid$(p$,j,3)):ifz=0thenz$=left$(b$,1):goto6890
6810 ifz+.5>10^lthenz$="left$("***************,1):goto6890
6820 z$=right$(b$+str$(z),l)
6890 j=j+3:lk=k:print#f," "z$;:return
6900 iflen(p$)<jthenj=0:return
6905 ifmid$(p$,j,1)="c"thenprint#f,"C";:return
6910 print#f,"1";:return
8000 print""left$(v$,24)"Push R and then Return to see the menu.
8005 print" Do you want to delete all students.
8010 print" Type YES or NO and push Return."
8020 gosub50005:ifs$="NO"goto8100
8025 ifs$="R"goto10
8030 ifs$<>"YES"thenprint""b$"":goto8020
8040 printleft$(v$,12)" Are you sure? Type YES or NO.
8050 print" Then push Return.
8060 gosub50005:ifs$="NO"goto8100
8062 ifs$="R"goto10
8065 ifs$<>"YES"thenprint""b$"":goto8060
8067 print"Deleting Class":td=ti+180
8070 print#15,"s0:old pupil file",d0:print#15,"r0:old pupil file=pupil file"
8072 input#15,en,em$:ifen>18thengosub10000:goto8070
8080 n=-1:p$(1)=""
8085 iftd>tigoto8085
8095 goto10
8100 print""left$(v$,24)"Push R and then Return to see the menu.
8105 print" Enter the name of the student you want
8110 print" to delete.
8120 print"Please type the student's first name":gosub50000:n$=left$(s$+b$,15
)
8130 ifs$="R"goto10
8140 print"Now, type the student's last name":gosub50000:n$=left$(s$+b$,15)+n$
8160 forj=1ton:ifn$>left$(p$(j),30)thennext:goto8180
8170 p=j:j=n:next:ifn$=left$(p$(p),30)thenf=3:d$="":goto8200
8180 printleft$(v$,24)"This name is not in the file.
8185 print"Press RETURN to enter another name.
8190 geta$:ifa$=chr$(13)ora$=chr$(141)goto8100
8195 goto8190
8200 print"Deleting student record.":td=ti+180
8220 forj=p+1ton:p$(j-1)=p$(j):next:n=n-1:dy=1:ifn=0thenn=-1
8230 iftd>tigoto8230
```

```
8240 soto10
9000 ifdy=0soto9010
9002 oren8,8,8,"00:pupil file,s,w":forj=1ton:print#8,p$(j)chr$(13);:next
9004 input#15,en,em$:ifen>18thensosub10000:close8:soto9002
9006 close8
9010 print"■■■■■■■■■     Reporting Program Terminated■■■■■■■■■■■■
9999 end
10000 print"■■■disk error■:  "em$:print"■Correct the problem.
10010 print"■Then press the shift key.":wait152,1:return
50000 print"and then press the RETURN key.■
50005 s$="":l=0
50010 geta$:s=asc(a$+z$):ifs>63ands<91thens=s+128
50020 ifs>192ands<219andl<15thena$=chr$(s):s$=s$+a$:l=l+1:printa$;:soto50010
50030 ifland(s=13ors=141)thenprint:return
50040 ifs=148ors=20thenifl thenprint"■";:s$=left$(s$,l-1):l=l-1
50050 soto50010

ready.
```

THE SWRL ADDRESSING ENVELOPES PROGRAM

Operating Manual

Joseph Lawlor

CONTENTS

PROGRAM OVERVIEW

The SWRL Addressing Envelopes Program teaches the capitalization, punctuation, and format skills needed to address envelopes correctly. The program is designed to be used by students in grades 3 and 4. The program includes two related systems: (1) the instructional system, which is used by students, and (2) the reporting system, which is used by the teacher.

The instructional system consists of three lessons that cover the following skills:

- Placing the address and return address in the correct space on an envelope
- Capitalizing the names of persons
- Capitalizing personal titles (e.g., Mr.)
- Capitalizing the names of streets, cities, and states
- Using a comma to separate city and state.

The instructional system provides instruction, practice, and remediation, and records student performance. Lessons 2 and 3 include review and reinforcement of skills presented earlier.

The reporting system allows the teacher to review student performance, delete student records when necessary, and print hard-copy reports for individual students or for the entire class. Up to 100 student records can be stored on the program disk.

HARDWARE REQUIREMENTS

In order to run the SWRL Addressing Envelopes Program, you must have the following hardware components:

1. A 32K Commodore PET computer equipped with a graphics keyboard.
2. 4.0 ROM or Upgrade ROM (often referred to as 2.0 or 3.0 ROM).
3. One Commodore 2031 or 4040 disk drive.

You may also wish to have a Commodore printer connected to the computer so that you can take full advantage of the reporting system. (Other brands of printers will not work properly with this program unless an interface has been installed to convert the Commodore character set.) However, the program will run with no printer connected.

USING THE INSTRUCTIONAL SYSTEM

Loading

To load the SWRL Addressing Envelopes Program, use one of the two procedures outlined below, depending on the ROM Version that your computer has. Before you load the program, check to be sure that the program disk does not have a write-protect tab on it.

- If your PET has Upgrade ROM, follow these steps:
  1. Turn the computer on.
  2. Insert the program disk into drive 0 and close the door.
  3. Type LOAD "*",8 and push RETURN.
  4. When the program has loaded and READY appears on the screen, type RUN and push RETURN.
- If your PET has 4.0 ROM, follow these steps:
  1. Turn the computer on.
  2. Insert the program disk into drive 0 and close the door.
  3. Hold down the shift key and press the key marked RUN/STOP. The program will load and run automatically.

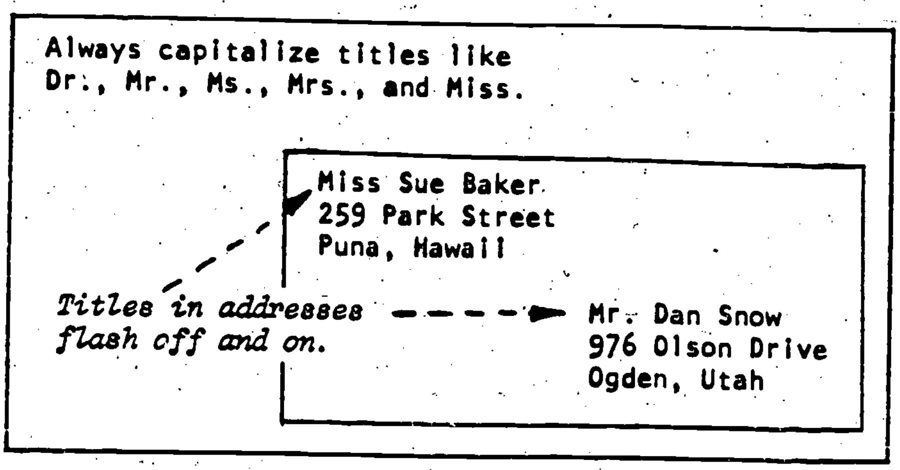
After the program has started running, leave the program disk in drive 0. The computer must have access to the disk to keep records and read data as the program runs.

Introduction

When the program has finished loading, a title screen appears. After a short delay, the title screen clears and you are asked to enter your first and last names. (The shift key has no effect in this program. Your name will appear in capital letters.) When you use the program for the first time, instruction begins with Lesson 1. If you have used the program before, instruction begins at the beginning of the lesson you were working on at your last session.

Instruction and Practice

The program provides instruction on specific rules for addressing envelopes. Each rule is presented, and examples are highlighted on the screen; e.g.:



Because of space limitations on the screen, the examples in the program do not include zip codes. In addition, all examples use full state names (e.g., Utah) rather than two-letter Postal Service abbreviations (e.g., UT). Although such abbreviations are commonly used on envelopes, they do not provide a convenient way to practice an important skill taught in this program--capitalizing the first letter in names of states. Consequently, teachers may wish to discuss the use of zip codes and abbreviations after students have completed the program.

Next, the program provides practice items for the rule that has just been presented. Directions for each item are displayed, and the program prompts you to respond by displaying your first name, followed by a question mark. Two types of items are used, one for address placement (in Lessons 1 and 2), and the other for capitalization and punctuation (in all three lessons).

In the address placement items, an envelope is displayed containing either an address or a return address. You are asked to identify the correct answer by pressing either 1 or 2; e.g.:

```
Which is on this envelope?

     1.  address
     2.  return address

Type the number for the right answer.

Joe?

                    Betty Gomez
                    857 First Avenue
                    Price, Idaho
```

If your answer is correct, the program continues with the next activity in sequence. If your answer is incorrect, the program gives you the correct answer and displays the rule and examples again. However, if you answer three consecutive items incorrectly, the program ends the lesson (see Exiting, below).

In the capitalization and punctuation items, you are asked to correct errors in the addresses on an envelope. A square, blinking cursor appears in the upper left-hand corner of the envelope; e.g.:

Move the cursor
and fix the names
on this envelope.

Joe?

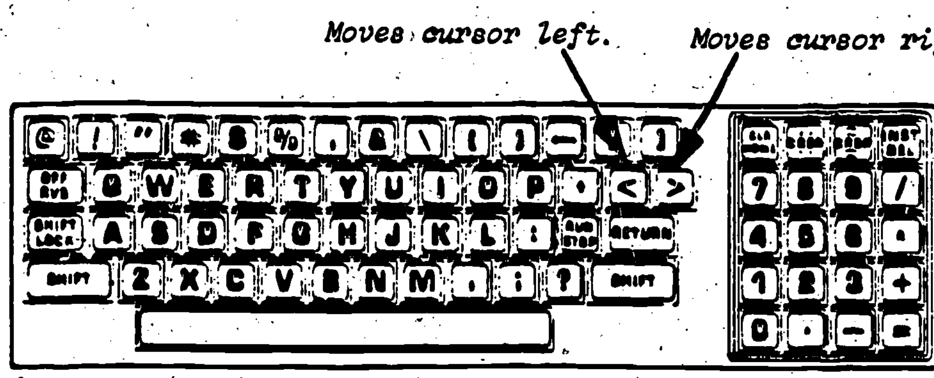
cursor

Mr. paul west
111 Maple Street
Anchor, Maine

Miss carol morris
1853 Arden Way
Post, Nevada

To correct the errors, follow the procedures below:

1. Use the angle bracket keys ('<' and '>') to move the cursor to the first error on the envelope. The position of these keys is noted in the illustration below:



You may hold down the appropriate key to keep the cursor moving continuously.

2. When the cursor is positioned on the error, press the letter or punctuation mark that should be changed or added. (You do not need to use the shift key to change a letter to a capital.) If your response is correct, the change is displayed on the screen. If your response is incorrect, the screen does not change. The program accepts only the correct changes in the correct positions on the envelope.

3. Continue procedures 1 and 2 until you have corrected all the errors on the envelope. When the errors are all corrected, you are informed of your progress, and the border of the envelope flashes.

If you do not correct all the errors within approximately one minute, you are told to continue looking for errors. If you still do not correct all the errors, the program makes the corrections and displays the rule and examples again. However, if you fail to make all the necessary corrections on two consecutive envelopes, the program ends the lesson.

Exiting

You may exit the instructional system for several different reasons, which are explained below. However, exiting the system does not end the program. After you exit the instructional system, the program returns to the initial title screen and starts over from the beginning. In order to end the program, you must enter the reporting system, which is explained on page 8.

The instructional system terminates when one of the following conditions occur:

1. You have completed an entire lesson successfully.

2. You have failed to push RETURN within one minute of being asked to do so.

3. You have answered three consecutive address-placement items incorrectly.

4. You have failed to correct all the errors in two consecutive capitalization/punctuation displays.

USING THE REPORTING SYSTEM

Introduction

When you have exited the instructional system (or when the program has just finished loading), a title screen appears, followed by a request for you to type your first name. To enter the reporting system, type the word REPORTS at this point. Typing anything other than REPORTS will cause the program to branch to the instructional system.

The Menu

When you enter the reporting system, the following "menu" of options is displayed:

Menu	
List students in file.	Push 1
Display Student Reports.	Push 2
Print Class Record.	Push 3
Delete a record.	Push 4
Exit Reporting System.	Push 5

Select the option you want by pushing the appropriate number. Each choice is explained below.

- List students in file. Selecting this option allows you to view the names of all students who have used the program.* You may also print a hard copy of this list if you have a printer connected to your system.

- Display Student Reports. This selection allows you to review the progress of each student on file. The Student Report shows which lessons the student has attempted and completed. In addition, performance on different sections within lessons is noted. A sample Student Report is illustrated below:

*Three sample student records have been included on this copy of the program disk. If you wish, you can delete these records by selecting option 4.

DAVID JONES

LESSON 1: completed

Address placement:	2 repeats
Capitalizing titles:	perfect
Capitalizing names:	1 repeat

LESSON 2: completed

Address placement:	perfect
Capitalizing titles and names:	perfect
Capitalizing streets, cities, and states:	1 repeat

LESSON 3: completed

Capitalizing titles and names:	perfect
Capitalizing streets, cities, and states:	perfect
Adding comma between city and state:	1 repeat
Review:	2 repeats

This sample Report shows that the student has completed all three lessons. The Report also indicates the sections of each lesson on which the student appeared to have some difficulty. For example, in the address placement section of Lesson 1, the student had to repeat two items before completing the section successfully.

If you have a printer connected to your system, you can print a single Student Report or Reports for all students in the class.

- Print Class Record. This option allows you to print an alphabetized list of all students in the class, along with a summary of their individual performance data. A sample class Record appears below.

KEY: L = Lesson; C = Complete; I = Incomplete; P = Placement; T = Titles; N = Names; St = Streets; Ci = Cities; Sa = States; Co = Commas; R = Review

CLASS RECORD

No. of Repeats

STUDENT	L 1	P	T	N	L 2	P	T/N	St/Ci/Sa	L 3	T/N	St/Ci/Sa	Co	R
1 Joan Best	C												
2 Thomas Frame	C				C				I	3	2	5	2
3 David Jones	C	2		1	C			1	C			1	2
4 Carol Kane	C				C				C				

This sample shows the performance of four students. The first student has completed the first lesson only; she has not yet worked on Lesson 2. The second student has completed the first two lessons, but he appears to be having some difficulty with Lesson 3, as is indicated by the number of repeats in each section. The third and fourth students on this Class Record have completed all three lessons.

In order to use this option, you must have a printer connected to your computer. The Class Record is not displayed on the video monitor.

- Delete a record. This option allows you to (1) delete all student records on the program disk, or (2) delete individual student records.

- Exit Reporting System. This selection allows you to leave the Reporting System and end the program.

When you select an option from the menu, the computer provides instructions on how to proceed. After you have completed a task (for example, printing a Student Report), the program either asks if you wish to perform an additional task within that option (for example, print another Student Report) or returns you to the menu to select another option. Note that you must return to the menu to end the program by selecting option 5.

SWRL COMPUTER-BASED INSTRUCTION
ON ADDRESSING ENVELOPES

Pilot Study

Ann Humes

SWRL COMPUTER-BASED INSTRUCTION ON ADDRESSING ENVELOPES

Pilot Study

One part of the NIE Communication Skills project is "Computer Utilization in Composition Instruction." The purpose of this inquiry is to investigate the use of microcomputers to improve composition instruction in the schools. One way to improve composition instruction is to provide appropriate primary instruction and practice and appropriate supplementary practice on skills that would otherwise require considerable teacher time. When students work with these skills on the microcomputer, teachers can devote more of their time to teaching higher-level elements of the composing process. Furthermore, using the microcomputer motivates students to learn because of the game-like nature of the medium.

The computer instruction on addressing envelopes, developed by SWRL staff, employs the game-like nature of the computer to present a set of related skills in a relevant context. The following skills are taught:*

- Places the address and return address in the correct space on an envelope.
- Capitalizes titles (e.g., Dr., Ms.).
- Capitalizes names of persons.
- Capitalizes names of streets, cities, states.
- Uses a comma between the city and state.

The specifications for computer-based instruction on addressing envelopes are detailed in an earlier section. This instruction is designed to be appropriate for third and fourth grade students.

---

*Postal service abbreviations are not taught because they contrast with the more general skill of capitalizing the names of states; zip codes are not taught because displaying them on the screen would preclude displaying the longer names of some states. Thus students would receive practice capitalizing only a few short names for states.

The instructional program is comprised of three lessons. In the first lesson, the computer displays a simulated envelope and instructs students on the placement of the address and return address. Then students practice identifying address and return address placement. Next the computer provides instruction on capitalizing personal titles, again in the context of an envelope. Practice on this skill is followed by instruction and practice on capitalizing the names of people. Students repeat each section of the lesson until they successfully correct all the errors in capitalization on each displayed envelope. The computer uses displays with new items for the repeated instruction and practice.

Successful students proceed to Lesson 2, which briefly reviews the skills presented in Lesson 1 before providing instruction and practice on another skill necessary for addressing envelopes: capitalizing the names of streets, cities, states. As in Lesson 1, skills are taught in the context of an envelope, and students proceed in the program according to their performances.

Lesson 3 is similar in design. Students first review the capitalizing skills presented in previous lessons. Then students are given instruction and practice on one additional rule that they must use to address an envelope: using commas between the names of cities and states.

The computer keeps records of the students' performances, which are available to the teacher through the reporting system. This system can provide a class record as well as detailed reports on individual students. It also provides a list of all students who use the program and allows the teacher to erase records of individual students or a whole class.

Programming for this instruction was done on a Commodore Pet by Jerry Bailey. After initial programming was complete, the instruction was used by staff to locate any errors in the displays or "bugs" in the programming. Staff found that the program ran very well and did what it was expected to do. A few typographical errors were discovered, and these were corrected. Additionally, the timing of some displays was adjusted so that the time allowed to complete individual practice items was more appropriate.

After the changes were made in the program, a pilot study was conducted to assess how well the program worked with users unfamiliar with its purpose and functioning. The pilot study involved three SWRL secretaries, adults who had no prior experience with the program. The results of the study are reported below.

Procedures

Three SWRL secretaries completed individually the instruction on addressing envelopes. They also worked with the reporting system designed for teachers. All three secretaries were experienced operators of the SWRL Word-11 word-processing system. Consequently, they were accustomed to working on a computer keyboard and with a monitor.

Before they worked with the Addressing Envelopes program, secretaries were informed of the grade level and topic of the instruction. They were also asked to make comments, if they wished, about features of the program. After they completed the instruction, secretaries worked through the reporting system for teachers. As secretaries sat at the computer terminal, a composition staff member sat with them to provide instruction when needed, to answer questions, and to observe.

Results: Instruction

No major problems occurred. A few minor difficulties with computer operation were noted. One problem occurred for all three secretaries: they did not understand how to manipulate the cursor until they were informed that they should use the angle-bracket keys. This confusion occurred despite the labels and arrows printed on paper and taped just at the top of the keyboard. Two secretaries expressed difficulty reading the small screen.

One secretary also commented that the flashing that occurs when the computer is correcting items is difficult to see--that it wasn't bright enough. Another flashing feature also caused comment: One secretary was disturbed by the brief flashing of the statement that all errors have not been corrected. She wanted to read the new text on the screen, but could not until the flashing stopped; this delay disturbed her. However, the flashing serves its function of attracting attention to the new element on the screen.

One secretary at first did not understand when she was to correct errors. The program includes an explanation that whenever the user's name appears, followed by a question mark, the computer is waiting for input. However, this message was not clear to the one secretary.

All secretaries commented that game-like aspects of the computer instruction would make the program enjoyable for its target audience of elementary school children. They particularly liked the flashing graphics (i.e., asterisks around the border of the envelope) that reinforces correct answers.

Results: Reporting System

Secretaries had no difficulty getting into the reporting system by typing "Reports" at the beginning of the program at the point where the computer asks for the student's first name. Each secretary tried out various features of the system by selecting one of the options on the main menu. All options functioned properly, and secretaries had no difficulty understanding the procedures. All secretaries chose to receive a printout of the student report on their own performance.

Conclusions

This small study indicated that both the instruction and the reporting system for the Addressing Envelopes program can be used with adults. Secretaries could handle all aspects of the program after they understood cursor movement. This information appears in the Operating Manual. Thus it appears that adults should be able to use the program without supervision when they have an Operating Manual. Difficulties in viewing the small screen are hardware problems that cannot be ameliorated by revisions in the program.

An appropriate further step in the development of the program would be a pilot study with users who reflect the actual audience for which the instruction was intended: third and fourth grade students.

APPENDICES

APPENDICES

A. Computer Instruction in Nearby School Districts

B. Professional Meeting Reports

C. Book Review

D. Courseware Reviews

E. Microcomputer Systems: A Tool for Instruction

F. A Proposal for an Elementary School Composition Program

G. Computers, Word Processors, and Composition Instruction

Appendix A

SWRL

SOUTHWEST REGIONAL LABORATORY TECHNICAL NOTE

DATE: July 10, 1980

NO: 2-80/06

COMPUTER INSTRUCTION IN NEARBY SCHOOL DISTRICTS

William Russell

ABSTRACT

In May and June, SWRL staff completed an informal survey of nearby school districts in connection with its microcomputer-based composition instruction project. Questions were asked about the districts' use of computer-based instruction in general and about microcomputers and composition instruction in particular. The survey did not cover all of the districts in any particular area nor was it intended as a representative sample of any area. Its main purpose was to identify schools for SWRL to work with in developing microcomputer-based composition instruction.

COMPUTER INSTRUCTION IN NEARBY SCHOOL DISTRICTS

William Russell

In May and June, SWRL staff completed an informal survey of nearby school districts in connection with its microcomputer-based composition instruction project. Questions were asked about the districts' use of computer-based instruction in general and about microcomputers and composition instruction in particular. The survey did not cover all of the districts in any particular area nor was it intended as a representative sample of any area. Its main purpose was to identify schools for SWRL to work with in developing microcomputer-based composition instruction.

We felt that it was important, particularly at this time of the school year, to be as simple and brief as possible and to politely terminate any conversation that did not appear to stimulate genuine interest on the part of the district staff member. For this reason we have widely varying amounts of information from these districts. There were two distinct telephone protocols, depending on whether the call was made by an AMPS or by a MPS. After introductions, both types of call began by simply determining whether the district currently used (or planned to use) computer-based instruction. If so, and if the district staff member showed no reluctance to cooperate, the two types of call proceeded differently. The call made by the AMPS followed a script (included here as Appendix A). The AMPS caller asked whether the contact would be able at this time to complete a brief mail questionnaire about the district's use or planned use of computers in instruction. The MPS caller, on the other hand, developed what information he could by phone. The results of these inquiries are summarized in here. Source data details are on file at SWRL.

calls are identified as either "telephone contact using script" or simply "questionnaire," depending on whether a follow-up questionnaire was sent. The MPS calls are labeled "full inquiry phone call."

We contacted thirty-five local districts, twenty-one in Orange County, twelve in Los Angeles County, and two in San Bernardino County. Nineteen of these were contacted by the AMPS, and fourteen of these AMPS contacts were followed up by questionnaires because they said they currently used or planned to use computers in instruction. (The questionnaire is included here as Appendix B). Follow-up calls to districts that did not reply to the questionnaire indicated that many of them had not completely understood the original phone call and actually did not have any current or planned computer-based instruction to report on.

The MPS made sixteen of the thirty-five contacts and pursued them according to the degree of district interest and the potential for the SWRL project. This led in some cases to conversations with school-level personnel and in two cases to visits to schools.

The information developed in these calls focused on identifying districts that the microcomputer-based composition project might work with during the 1980-81 academic year. In order to select one or more districts to work with, it was useful to know the type of equipment they had, the current focus of their computer instruction, and the general level of sophistication of their programs. Schools with microcomputers were of particular interest, because that is the emphasis of the project. For the same reason any computer-based composition instruction was of interest, although almost none was encountered. Other information would have future value and, it was hoped, would lead to some useful inferences about the general state of the art of computer-based instruction.

District administrators were not always aware of small computer instruction projects within their district and in at least a few instances they did not understand from the first phone call that we were inquiring about student use of computers in instruction as opposed to other uses of computers in the district. With those qualifications, we found that twelve of the thirty-five districts were current users of computers for instruction and seven had plans for using computers for instruction. Several districts volunteered that it was the lack of funds rather than a lack of interest that prevented them from using computers for instruction.

In all cases where we suggested the possibility of future cooperation, the district administrator or resource teacher was more than willing. Our problem in this respect is only that there are very few districts that have the resources for a tryout of microcomputer-based composition instruction. Los Angeles Unified School District appears to be an exception, and we have agreed to talk with them again in late August about a possible school to work with next year.

We felt that it would also be valuable to the project to have some tryout experience with a smaller district. Of the districts we contacted, the only districts (large or small) that had microcomputers were the following:

Claremont Unified (10 Heathkits used in computer science)
Beverly Hills (1 Apple, use unspecified)
Los Angeles Unified (about 200, mostly TRS-80s, various uses)
Westminster Elementary (2 TRS-80s, math and computer science)
Saddleback (unidentified, at one junior high school)
Arrowview Intermediate School, San Bernardino City Unified
(a Nestar Corp. microcomputer network, use unspecified)

Districts that used time-shared terminals rather than microcomputers (or in addition to them) were:

Beverly Hills (33 terminals)

Newport-Mesa Unified (an extensive DEC-based system)

Huntington Beach Union High School (an extensive IBM-based system)

Los Nietos Elementary (a commercial system with 48 terminals)

Placentia Unified (four terminals)

Los Angeles Unified (various systems)

Arcadia Unified (one terminal)

Long Beach Unified (a math demonstration program at Franklin Junior High School)

Most of these latter districts had older and more extensive programs covering a wide variety of subjects, although only one, Placentia, reported that students type and revise compositions using a terminal. Placentia indicated that they had just four terminals (not microcomputers) and that the terminals are also used by K-12 students for other subjects, so use for composition instruction could not be very extensive.

If we decide to work with a smaller district as well as with Los Angeles Unified, Westminster is close and the teacher would be more than willing. Her resources are very limited, however. Arrowview School in San Bernardino is of some interest because of the more extensive microcomputer system that it has, but it has not been approached about possible cooperation. Another possibility is Yucaipa, of interest because they have a large creative writing grant and plan to use a word processor in conjunction with their project. At present they do not envision student use of the computer, but appear to be open to suggestions, especially if we prove to be of help to them in getting their project started.

As noted earlier, we do not claim that the survey reported here is a representative sample. Nevertheless, our findings are consistent with those of a more comprehensive survey that was done in another area. A more formal survey of computer-based instruction was conducted in the Northwest by the Northwest Regional Educational Laboratory in early 1979 (EDUCATIONAL COMPUTING IN THE NORTHWEST, Portland, Oregon, NWREL, June 1979). Eight hundred ninety-two postcards were sent to superintendents in six states (Alaska, Hawaii, Washington, Oregon, Montana, and Idaho). Three hundred ninety-nine usable responses (44.73%) were received. Of these, one hundred thirty-six (34%) were currently using computers for instruction. This is the same percentage found in our own survey (12 of 35 districts).

In the course of our survey, we developed some impressions which turned out to be consistent with another part of the NWREL survey, a survey of teacher needs in the same six states. The NWREL findings can be found in the NWREL document mentioned earlier. Our overall impression of the state of the art of computer-based education in the districts we talked with is that it is in its infancy, that minicomputer systems have more established, systematic programs, but that microcomputer systems are gaining ground because they are less expensive and more flexible. In most districts the focus of interest is on mathematics and computer science. Outside these subjects, courseware is primitive. When we talked to teachers or other personnel directly involved in computer-based instruction, they were aware of the lack of sophisticated courseware and impatient for the potential of this medium to be realized. This is not to say that they were generally unhappy with current practices. In most cases they were very enthusiastic. It is our impression that any SWRL suggestions for extending and improving courseware will be well received by current users, but that they will not be quick to give up practices with which they are familiar and generally happy.

Attitudes of personnel not currently involved in computer-based instruction are quite different. While there are teachers or administrators who would like to have computer-based instruction for particular applications, there does not appear to be much interest in computer-based instructional systems in general except on the part of current users and a few enthusiasts, usually teachers rather than administrators. At the district level, computers are more frequently thought of for their potential as tools in record keeping and instructional management systems. The NWREL survey found that the need for computer-based instruction was most often felt in mathematics and computer science, more often at the high school level than at elementary schools. This was also our impression.

APPENDIX A

SCRIPT FOR USE BY AMPS WHEN MAKING TELEPHONE CALLS TO DISTRICT OFFICES

KEY: A. You say the capitalized phrases.

B. The sentences in normal upper and lower case are instructions, not to be spoken.

C. Dashed lines (------) indicate that a name or title of address would be appropriate to say at this point.

Most of your calls will involve two steps: first a conversation with a secretary and second one or more conversations with district administrators. The script that follows is divided into these two steps.

STEP ONE

(Your first interchange with whoever answers the phone)

1. I'M CALLING TO ASK IF YOUR DISTRICT USES COMPUTERS FOR INSTRUCTION.

   If you already have a name: WOULD ------ BE THE PERSON TO TALK WITH?

   If you don't have a name: WHO WOULD BE THE BEST PERSON TO TALK WITH ABOUT THIS? Determine this person's title of address (Mr./Ms./Dr./etc.) as well as the name.

2. If you don't reach an appropriate person, make a note to call back.

STEP TWO

1. MY NAME IS ------ --------. I'M WITH SOUTHWEST REGIONAL LABORATORY. WE'RE A NON-PROFIT NATIONAL EDUCATION RESEARCH FACILITY LOCATED IN ORANGE COUNTY. MAYBE YOU KNOW ABOUT US?

   (Respond briefly to any questions or comments about SWRL, then continue.)

2. ARE YOU THE PERSON TO ASK ABOUT COMPUTER INSTRUCTION IN YOUR DISTRICT?

   If it is clear at this point that the district has no computer instruction, politely terminate the call unless your contact shows further interest.

If this contact refers you to another person, start again at STEP TWO above, mentioning the person who referred you.

If your contact remains interested, continue with point 3. below.

3. WE HAVE JUST BEGUN A THREE YEAR NATIONAL INSTITUTE OF EDUCATION CONTRACT TO DEVELOP COMPUTER BASED INSTRUCTION. AS A PART OF THIS PROJECT, WE ARE DETERMINING WHAT COMPUTER HARDWARE AND WHAT TYPES OF COMPUTER BASED INSTRUCTION ARE ALREADY IN USE IN THIS REGION. DOES YOUR DISTRICT USE OR PLAN TO USE COMPUTERS IN ANY OF ITS INSTRUCTIONAL PROGRAMS?

If no: THANK YOU VERY MUCH FOR YOUR TIME, -----. End of call.

If yes, continue.

4. WE WOULD LIKE TO MAIL YOU A SHORT QUESTIONNAIRE ABOUT YOUR COMPUTER BASED INSTRUCTION. IF THIS IS OK, SHOULD IT BE MAILED TO YOU OR WOULD IT BE BEST TO SEND IT TO SOMEONE ELSE IN THE DISTRICT, OR POSSIBLY TO MORE THAN ONE PERSON?

If there is any reluctance to accept the questionnaire: IF THIS IS NOT A GOOD TIME TO SEND IT, WE WILL NOT DO SO.

If the reluctance continues: THANK YOU VERY MUCH FOR YOUR TIME, -----. End of call.

If your contact is agreeable, write down the title, name(s) and address(es) of the recipient(s) of the questionnaire.

5. THANK YOU VERY MUCH FOR YOUR TIME, ------------. End of call.

Survey page 2

3) Indicate the approximate number of student terminals of each type listed below that are currently in use in your district. Where you know of plans to acquire such terminals, please write the approximate number to be acquired followed by the word "planned."

"Dumb" terminals connected to a central computer (i.e., terminals that can't function by themselves as independent microcomputers) ______

"Smart" terminals standing alone (e.g., microcomputers such as those made by Apple, Pet and Radio Shack) ______ Please write the brand names here if you know them.

"Smart" terminals connected together ______ Please list brands above and circle them.

Other (describe)

Comments:

4. How do students enter responses or other data? Check the device(s) they use and write "planned" for hardware you plan to acquire.

Typewriter keyboard ______ Light pen ______ Punched cards ______

Numeric pad (similar to a calculator keyboard) ______

Bar code reader ______ Optical character reader ______

Graphics board ______

Other (describe)

Comments:

Appendix B

APPENDIX B

PROFESSIONAL MEETING REPORTS

NATIONAL EDUCATIONAL COMPUTING CONFERENCE

DENTON, TEXAS, JUNE 1981

Ann Humes

NECC was a valuable experience because it helped me become more familiar with CAI issues, although we should have acquired some of that knowledge much sooner. Nevertheless, I'm convinced that we know more about good instruction then nearly anyone who is currently developing CAI.

Coping with NECC logistics didn't allow much time for encounter; however, I did meet a few people. One was Alfred Bork, the CAI icon from the Educational Technology Center at Irvine. He is interested in getting together with us at a time when his entire staff is available. He mentioned that he worked with Northwest Regional Lab at one time and seems quite interested in working with SWRL. He has been aware of SWRL because he explained to his workshop attendees that SWRL is an NIE-funded laboratory predating the formation of NIE. He added this information after I introduced myself at the beginning of the workshop (he had each attendee identify him or herself). Bork is interested in doing something with composition, and they at ETC have thought about it "from time to time," but so far they haven't developed anything.

I also met Bob Shostak from Florida, who covers language arts for The Computing Teacher. He wants me to send him information about our project, and he'll put it in his column in a fall issue.

Attendance at NECC has nearly doubled over the last year, yet still very little was offered in our area, despite the apparent interest in CAI. The only presentations on the program that were even close to the composing or to the kinds of language-arts activities we are involved in were (1) Mike Southwell's grammar instruction and (2) Hugh Burns invention program.

A couple of themes were apparent throughout the conference. One was that Pascal is rapidly displacing Basic, although Basic still is the most widely used programming language. The innovators in the field are definitely in the UCSD Pascal camp. I heard Pilot mentioned only once--when Bork said that he didn't like such authoring languages. The trend toward programming in Pascal suggests that someone on our staff should develop expertise in that language. Another theme was that there is a pervasive concern over computer games because most of them do not entail any real mental or instructional activity and because they are perceived as sexist and violent.

Alfred Bork, Educational Technology Center, University of California at Irvine, Computer-based learning pre-conference workshop

Bork had each person in the workshop introduce him or herself and then he, in turn, commented on the Educational Technology Center. He noted that the ETC was originally involved in time sharing, but the Center has recently become interested in personal computers, physics, statistics at the college level, formal reasoning, and math competency.

Bork overviewed the "fourth revolution" in education. From 1990 to about 2000 A.D., he foresees rapid introduction of microcomputers; after 2000 he expects extensive use of microcomputers, with the public being unaware that a microcomputer revolution had occurred. Additionally, many self-paced courses will be available, and these courses will not necessarily be conducted in formal schools. The courseware development may follow the Open University model--off-campus education. The language, according to Bork, will be either Pascal or Ada, and most machines will be "stand-alones." He further sees an emphasis on learning rather than on hardware; this emphasis will make the original "Renaissance Man" developer/programmer obsolete because one individual cannot have expertise as both a programmer and a developer since it takes 500 hours to become minimally proficient as an educational programmer in a general-purpose language (Bork does not like authoring languages like Pilot).

Bork acknowledges several important issues to be considered in designing computer instruction: the dichotomy between small-scale and large-scale production (with a corresponding move towards large-scale production); a choice between what is expedient now versus what is appropriate for the long term; a dichotomy between development hardware and delivery hardware; the use of single screens versus many screens; use of special languages versus general languages; and problems of transfer, particularly for situations with both development and delivery machines.

Bork then discussed the ETC authoring process, which entails pedagogical design, graphic design (structuring the screen in space and time), implementation, evaluation, and improvement. Bork reports that it takes four people a week of full-time effort to design (not completely develop) about one hour of student materials. Now this effort does not involve programming or flow charts other than a simple kind of flowchart (see attached Addendum). This four-person-week time is simply for brainstorming the ideas, organizing them, and determining the direction of the instruction (again see the content in the Addendum). These four people include about three or four teachers. Sometimes an individual from Bork's group helps structure what is produced. Bork does not include programmers in his development sessions, claiming that they are the least desirable kind of developers because they are driven by the technology rather than by the instruction.

This group poses some problems during the authoring process. For example, they frequently want to write the materials as if they were writing text-like material rather than interactive material, and they tend to take a linear approach, treating all learners in the same way rather than individualizing the computerized instruction. They usually spend too much time on in-depth analyses of student responses, yet, according to Bork, people's responses are not so varied. Thus, he says, artificial intelligence methods are not needed for instructional design because 90% of the expected responses can be determined without these methods. Finally, this group of developers generally overuses "fancy language."

Bork noted some important screen-design factors that stimulate interest and/or enhance readability:

1. Large amounts of blank space, with little text starting at the left margin. Bork stated that right as well as left justification is not a good idea. Furthermore, he asserted that some of the best materials have a ragged left.

2. Short lines.

3. Speed of text output. Sometimes the output is much slower than is possible or optimal for reading.

4. Absence of hyphenation.

5. Minimal scrolling. With too much scrolling the user is reading the bottom line all the time. Rather, windows can be established and the text can scroll within those windows. However, a one-line scroll should seldom be used.

6. Good graphics. Such graphics require specialized software, internally-operating software. (Bork said the ETC has developed its own graphics software.)

7. Upper and lower case combination. A combiniation is better than upper case alone.

8. Lines on the screen that do not break across phrases.

In continuing his discussion of the ETC development method, Bork stated that he uses UCSD Pascal because it's highly portable and is "the best development choice." He commented that the real problem for "moving" is not the language, but the size of the language. Bork then distributed flowcharts to explain the processes materials go through before they are finally disseminated. The flowcharts are included in the Addendum. The first chart displays the complete procedure, and the subsequent charts detail the individual blocks within the larger chart.

Bork then involved the conference participants in developing, as a group, a lesson designed to teach rapid estimation. He began this group effort with a brainstorming session, having everyone in the room contribute ideas for instruction while he wrote them on the board. Then he categorized the ideas by type--strategies for estimating information, ideas for structural design, visual information, ideas for organizing the program.

The following points reflect comments that Bork made during the production of the rapid-estimation material:

1. He commented on computer games, noting that there is great concern about their characteristic violence and about male dominance of the types. Furthermore, games are turning kids off to other uses of computers. Bork also expressed concern over the competitive nature of computer games.

2. He thinks that while it is good to tell students to press the return key in the early part of a lesson, this statement does not have to be repeated after the student is well into the instruction--it is extraneous material on the screen, and nothing on the screen should be unnecessary; material on the screen should be erased when it is not needed.

3. Different kinds of feedback are probably needed because of the different kinds of both work and people that developers must deal with. Graphical feedback can often be used rather than text feedback. Correspondingly, he cautioned that we should be careful with feedback algorithms--we need more random access of such reinforcers as "good" and "super."

Bork's groups develop all the instructional design and then everything is sent out for review. After review, instruction goes to graphic design, and then finally to the programmers. Bork gives the programmers a flowchart and goes over that chart with the programmer. The programmer doesn't get back to Bork (unless there is a problem) until a program is running. Then Bork and the programmer sit down and work out the problems.

The last part of the workshop was devoted to a summary of the history of education and the current and future developments in education. For the history of education, Bork cited first learning by experience, then in schools, by writing, from lectures, from Socratic dialogue, and then finally from textbooks. Some of the current developments in education he referenced are competency-based learning, testing to specific objectives, mastery learning, and "learning at a distance." This latter kind of education may occur at an open university (which uses more funds for course development than for delivery systems) and is very efficient and effective. Bork also foresees a serious decline in the number of students, with many of that reduced number involved in non-traditional universities such as the Open University. He also cited (1) the University of Mid-America in Lincoln, Nebraska, (2) Coastline College, (3) Wang University, which is a computer company's school, one that has developed a full-scale graduate program, and (4) "Holiday Inn" University. He says that such internal training programs are now staggering in size and number.

As future developments in computer technology, Bork sees voice output, music output, sound cartridges, improved graphics, improved text, local networks, total learning environments, intelligent videodiscs, full multi-media dialogues, voice input, and even response interpretations through brainwave reading. Bork expects much of the future production and distribution of computer material to come from universities, from centers, and from computer venders rather than from book publishers; publishers seem incapable of moving in innovative directions. He also predicts that completely new companies, such as Computer Curriculum Corporation, will be formed to produce CAI. Bork also foresees that institutional change will be effectuated by increased use of computers--changes such as more self-paced coursework, more self-paced curricula, changes in grading systems, new types of institutions, and long-distance learning.

The remainder of the session was devoted to questions and answers, most of which were addressed to the attendees' specific situations.

Dr. F. James Rutherford, Office of Educational Research and Improvement. Opening Session.

Rutherford discussed recent attacks on education, suspicion of teachers and educators, and the potential of technology for upgrading education. Technology can provide such things as enrichment and career information, and can do chores such as taking inventories and compiling directories. He added that computers can facilitate interesting individualized instruction, including mastery learning and diagnostic instruction, and can deal with formerly "impossible problems" in education (e.g., educating migrant children and people with language handicaps, coping with teacher shortages in math and sciences, shifting students' frames of reference through role playing in problematic situations, providing real databases, introducing new methodology).

Rutherford commented that computers may be the only means for provoking curriculum reform because the move to a stronger computer-based form of education will require a re-examination of the curriculum. However, he worries that this reformation won't occur because (1) we are so committed to book technology that we have forgotten other, earlier forms of technology, such as the oral method, and (2) there is much teacher resistance (because the goals of innovators and the goals of teachers are mismatched, thus making demands on teacher time and posing a threat to teachers' jobs). Furthermore, the system itself is a barrier because of its cost, the software problems, the hardware problems, and the required training.

A. Joseph Turner, Clemson University. Program Development Methodology.

Turner, who was repeating the tutorial he gave at a previous NECC, stated that program design entails (1) modules [a functional component of a program] and (2) top-down design [the hierarchical design that starts with a top box and each box is broken down at lower levels of hierarchy]. A top-down design is complete no matter where the hierarchical levels end: The hierarchy at step 1 consists of the top box and the boxes underneath that; at step 2, the hierarchy involves the boxes under the second row of boxes; step 3 adds the boxes under the previous row of boxes, etc.

A desirable characteristic of a module is a single entry-single exit, independent of the "outside world." The module doesn't "know" what is happening in the rest of the program and, therefore, is easy to pluck out and replace. A good module is also relatively small (e.g., 100 statements maximum). It performs a single logical function and is separately compiled. Specifying a module should include two kinds of statements: statements about (1) the function performed [what is the relationship between the input and the output--given the input, how are the outputs determined], and (2) the inputs and the outputs [the parameters, the arguments, any information from the outside].

Information-hiding is another feature of good modularization. In other words, data manipulated by a module should be "hidden" from other modules, and only one module should manipulate the same data. Modules should be clearly delineated by a new page or separated by blank lines; the description should specify the input and the output, the function and the use of "local variables," and the code used in the module. The program should be easily read and understood by others, and it doesn't have to use machine language.

Refining modules requires a top-down development of the code. The procedure includes (1) expressing the higher level or outer level steps of the module in code; (2) refining the steps by breaking up each high-level step into a sequence of more detailed [lower-level] steps; (3) continuing the refinement process until each step can be easily coded in the programming language. Guidelines for stepwise refining include postponing details as long as possible, and redesigning rather than patching previous work if flaws are found.

The remainder of the session was devoted to a very technical discussion of module coding and implementation of the top-down design.

Richard Paulock, Minnesota Educational Computing Consortium, Chair. Panel Discussion on Videodisc Research and Applications.

One of the three panel members was a substitute for Joan Sustik of the University of Iowa; his first name was Jim (I didn't catch his last name). Other panel members were Richard Brant of the University of Utah, and Alfred Bork of the University of California at Irvine. Notetaking for this session was difficult because panel members turned out the lights to present their black-and-white slides.

Richard Paulock, the chair, began the session by noting that the optical reflective videodisc is the only type of videodisc that is not ruined by handling. Other types include a transparent disc and the RCA disc. Videodiscs have good capabilities for stop and go, one frame at a time. Two different instructional tracks can be put on in the same unit, one remedial and one enrichment, but there is no audio with reverse. The videodisc can go in slow motion and can be made to pause.

Paulock said that to produce a videodisc, the developer should (1) start with a good curriculum; (2) determine the media interfaces (even paper and pencil can be a medium); (3) examine existing materials; (4) select artists and actors; (5) film in a studio (production costs at least $500 a minute, which is cheap, for $10,000 a minute is not out of range); (6) develop the computer program, including the story board, the script, and the interfacing with the microcomputer; (7) make the pre-master.

Joan Sustik's substitute, Jim _____, briefly recounted the history of the videodisc development and then discussed current projects that involve retrieving library information, searching art history, providing information on surgical procedures, and teaching ballet.

Richard Brant cautioned developers not to be swept away by technology and to remember that many applications of the videodiscs can be done as well and yet less expensively on videotape. He stated that videodisc technology is needed where high visual realism is very important, not where line drawings will do just as well. Brant added that if another medium or another technology would be just as effective, it should be used rather than videodisc. Currently the best instructional content for videodiscs is procedural knowledge (e.g., explaining military procedures; teaching ballet).

Creating generic videodiscs is important. A generic videodisc has no notations of computer code or organizational affiliation and is usable as material for other kinds of lessons. If developers make a practice of creating generic discs, videodisc will widely be used. Consequently, Brant and his group try to keep as much material as possible on floppy disks so that the videodisc material can be reused. Thus Brant disapproves of putting text to be read on the videodisc.

The last participant was Alfred Bork, who noted "five commandments" for people who work with videodiscs:

1. Emphasize learning and not technology. When developers become involved in technology, the technology rather than the instruction becomes dominant. Much of the earlier videodisc material evidences this problem.

2. Focus on long-range progress rather than on immediate success. According to Bork, the strength of his project is in its long-range view; early solutions are usually simplistic and should be rejected. He also feels strongly that people who work with videodiscs need to look at the whole curriculum and not just bits of it. Bork cautioned that research is needed on the videodisc medium and its capabilities, yet essentially nothing has been done. (He claims to have seen only one study.)

3. Remember that working with intelligent videodiscs requires blending a variety of media. Acceptable material won't be produced until developers understand how to blend these media; they can't just take bits from the computer and bits from the video. One frequently used strategy has involved putting previously filmed material on a videodisc, and Bork claimed that this has produced little first-rate material. Another strategy is to take good computer-based learning material and add a visual component. Although no one seems to be following this procedure now, Bork prefers it. However, the ideal procedure, according to Bork, is to develop fresh material--entirely new material on both the video and the computer. This method is most likely to get that perfect blend. A final qualification is that developers should always ask themselves whether what they're doing on the videodisc could be done better with other technology (and this technology may include paper and pencil/print).

4. Investigate full-scale production problems. Bork again referred to the Open University, as he had in the pre-conference workshop, asserting that the best curriculum materials come from there--that the Open University knows more about developing curriculum materials than any other establishment.

5. Examine what has already been done. Bork considers the best material to be the medical simulation videodisc produced at the University of Iowa. He also mentioned Bob Fuller from the University of Nebraska, who has developed something "very sensitive pedagogically." Bork, along with others, believes that procedural information is the best content for videodisc instruction. He predicts that videodiscs will be developed first for K-12 because there is a larger market at these levels and because these levels are evidencing greater problems.

Bork referred to the WYCAT study at Brigham Young University. Several other participants mentioned this study during the course of the convention. If my notes are correct on what Bork reported, students who used the videodisc material did better on the posttest; however, a second posttest, administered after a two-week interval, evidenced no differentiation between the students who used the videodisc and those who used regular materials.

Graphic Design Issues in Computer Based Education. Ken Modesit, Texas Instruments, Chair. Participants were Jessica Weissman, University of Delaware, and Brian Shankman, American Airlines, Euless, Texas.

Jessica Weissman first listed three uses for graphics: to simulate, to allow manipulation of pictures in order to understand concepts, and to illustrate. She added that lessons can be identified by matching the titles with the graphics, and that graphics can be used to motivate, to break up text, and to enliven long passages of text.

According to Weissman, the same type of material should be consistently positioned in one place on the screen; the text and the graphics should not be randomly exchanging places. She also said that developers need to think in terms of square rather than oblong space (as in paper or book instruction). She also cautioned against crowding the screen.

Weissman stated that the brightest part of the screen stands out for the viewer, and "brightened" material will have emphasis. Color, too, adds emphasis. Answer choices can be emphasized by being boxed. However, Weissman tempered her support of boxes by noting that they are a tempting snare because they can be so overdone that the user can't determine what is most important. She commented on other abuses of graphics, such as using different typefaces and "cutesy" pictures that serve no instructional purpose. Weissman warned developers not to overuse flashing; it should be used sparingly. Furthermore, the timing for flashing is crucial, so it must be carefully determined; otherwise the flashing may occur when the student is focusing on reading other material on the screen. She also cautioned against using graphics that may not be recognizable or realistic.

Pictures are preferrably oriented to the left of the screen and text to the right (purportedly because of left/right brain functions). A line dividing graphics from the text may aid comprehension of the material on the screen. Directions should be easy to find; they shouldn't be situated so that the student has to search the screen for them.

The final speaker, Brian Shankman, presented some graphics that American Airlines uses to teach pilots cockpit instrumentation. I stayed for only part of Shankman's talk because I wanted to catch Hugh Burns' presentation. According to the program, Burns should have been speaking concurrently with Shankman. However, Shankman did make one relevant point before I left: When the graphic element is placed in the upper-left of the screen, the pilot tends to ignore the rest of the screen.

Hugh Burns, United States Air Force Academy

Because one of the other presenters was absent, I caught only the end of Burns' presentation. The audience was questioning Burns when I slipped into the room.

In response to his questions, Burns stated (1) that his program appeared to have no effect on arrangement or style, (2) that a good writer pursues about five heuristic questions before writing his/her composition, (3) that the computer responses (e.g., "By George!") are obvious to students within 10 minutes (some students accept this obviousness and some don't; well-prepared students may even complain about these interruptions), (4) that getting started is the most difficult aspect of his students' composing processes, so his program is an asset in helping them get started.

Burns noted that the program is written in Basic, but he would like to rewrite in it in Pascal if he could find the time. He added that Denise McGinty, of the Nebraska Learning Center, has developed an Apple version of his program.

Computer Graphics in Education. Eugene Herman, Grinnell College, Chair. Participants were Alfred Bork, University of California at Irvine, Robert Myers (for the absent Cristina Hooper), University of California at Santa Cruz, and Arthur Luehrmann, Computer Literacy, Berkeley, California.

Alfred Bork commented that personal computers run graphics more easily than do main-frame computers, but their graphics potential isn't generally realized because most "faculty" are poor at visual design--they are verbally oriented. Thus teachers (the instructional designers for ETC) must be pushed to think visually. Bork categorizes all elements of the screen display as graphics. Thus each choice of text placement should be evaluated for graphics criteria because text placement affects readability. He then discussed readability, enumerating the same features as he did in the pre-conference workshop. He added that his ETC group uses graphic artists to develop graphics with their usual tools rather than with computer code. Not until these graphics are complete does he turn them over to professional programmers for coding.

After Bork's presentation, Robert Myers discussed graphics as components that can express concepts in a non-literal fashion, can present multiple views of the same thing, and can represent development over time. He added that graphics can be used to enhance memory of words, to get attention, and to display information in different ways simultaneously. The complexity of programming graphics increases as they move from static to sequence to animated to interactive. Compounding that complexity are the important pauses and pacing that must be carefully planned. Despite these programming problems, Myers cautioned, the medium must not be allowed to drive the instruction; the instruction must drive the medium.

Eugene Herman then discussed another big problem in graphic use--inadequate software. He asserted that what exists is too obscure, primitive, and varied. However, a graphic standard is now being developed and should be approved by 1983. Herman believes this standardization will make it easier to move programs from one computer to the other: Standardization will make programs device-independent and language-independent.

Arthur Luehrmann ended this session with a slide show--he made no informative comments about graphics, discussing only the specific ones on the screen.

Microsift Project in Courseware Evaluation.

Microsift is the NWRL clearinghouse for microcomputer applications in education. It collects, organizes, and disseminates relevant information. Each applicant is evaluated by professional teachers, and a Microsift staff member summarizes the review.

Robbie Plummer of the Region X Education Service Center of Richardson, Texas, was supposed to be the presenter. However, Plummer could not be there, so a local Richardson teacher, Sandy Maddox, appeared in his place. It was immediately evident that Plummer would not provide relevant information; she intended to talk only about the kinds of courseware being examined for use in her individual school district. Consequently, I left this session in search of something more relevant, but little of a general, non-technical nature had been scheduled for this time period. I finally decided to sit in on the Pascal tutorial, which was conducted by Harry Haiduk of Amarillo College. This session would have been more valuable if I had caught its beginning, but I do have a few notes on Pascal.

Pascal has 11 basic data types and 11 statements. The basic data types are used as building blocks. The composable statements consist of seven structured statements and four simple statements. Although GO TO is one of these statements, it isn't used much. According to Haiduk, a person who masters these 22 features (the 11 basic types and the 11 composable statements) and understands where they are used will then know Pascal.

Haiduk claimed that Pascal is easy to learn and to use. It has consistent syntax, consistent meaning, and it is fast and cheap. Furthermore, it compiles quickly and is error resistant because the compiler catches many of the errors. He listed the following features as those that catch errors: (1) All variables are declared and (2) there is only one automatic conversion. Pascal has inter-module communications (subroutines and functions) and is easy to read because it is as "English-like" as Basic. Haiduk discussed the specific data types and composable statements.

Network Information Resources for Computers in Teaching and Learning. Carl Zinn, University of Michigan, Chair. Other participants were James Johnson, CONDUIT, University of Iowa, and Joseph Lipson, National Science Foundation.

James Johnson described CONDUIT as an organization that reviews tests and distributes computer-based instructional materials. Most authors of current CONDUIT packages are university people who create materials based on their own coursework. I really didn't understand the purpose of Johnson's presentation. Instead of describing/explaining the Network, he talked about the need for and benefits of publishing, claiming that textbook publishers offer so much more than individual authors. According to Johnson, publishers offer full courses, mainlines, supplements, and distribution. Materials that publishers provide have been edited, carefully reviewed, polished, and subjected to assessment in the marketplace. He tempered his praise by adding that publishers' materials are developed for a mass market, are school rather than student oriented, and evidence little innovation because they must suit the needs of that mass market. Thus gaps must be filled by resource centers, clearinghouses, and meetings, and quasi-publishers like CONDUIT. Before accepting and distributing CAI, a publisher requires evidence of a large market, economy of scale (the cost can be spread across a large number of users), good authors, and knowledge of the medium. According to Johnson, publishers are currently exploring the possibility of changing the type of materials they are developing, particularly for K-12, but any change will probably be incompatible with the current school system.

Joseph Lipson discussed the role of federal agencies in CAI development. The federal government assists publishers by providing market information on whether ventures are feasible. The federal government supports a variety of dissemination efforts, builds networks across disciplines, and nurtures talent when a new kind of talent is needed.

Lipson charged the federal government with doing an inadequate job for several reasons. For one thing, many people who need information are not reached. ~~Additionally, information itself may not be useful because it~~ frequently must be changed before it can be used. Furthermore, people are now so inundated with information that they can't or won't sort it out. Another problem is that those who are supposed to serve are often reluctant to do so. Finally, the federal government has been reluctant to use the mass media to disseminate information because it doesn't want to appear to be either propagandizing or advertising for more money.

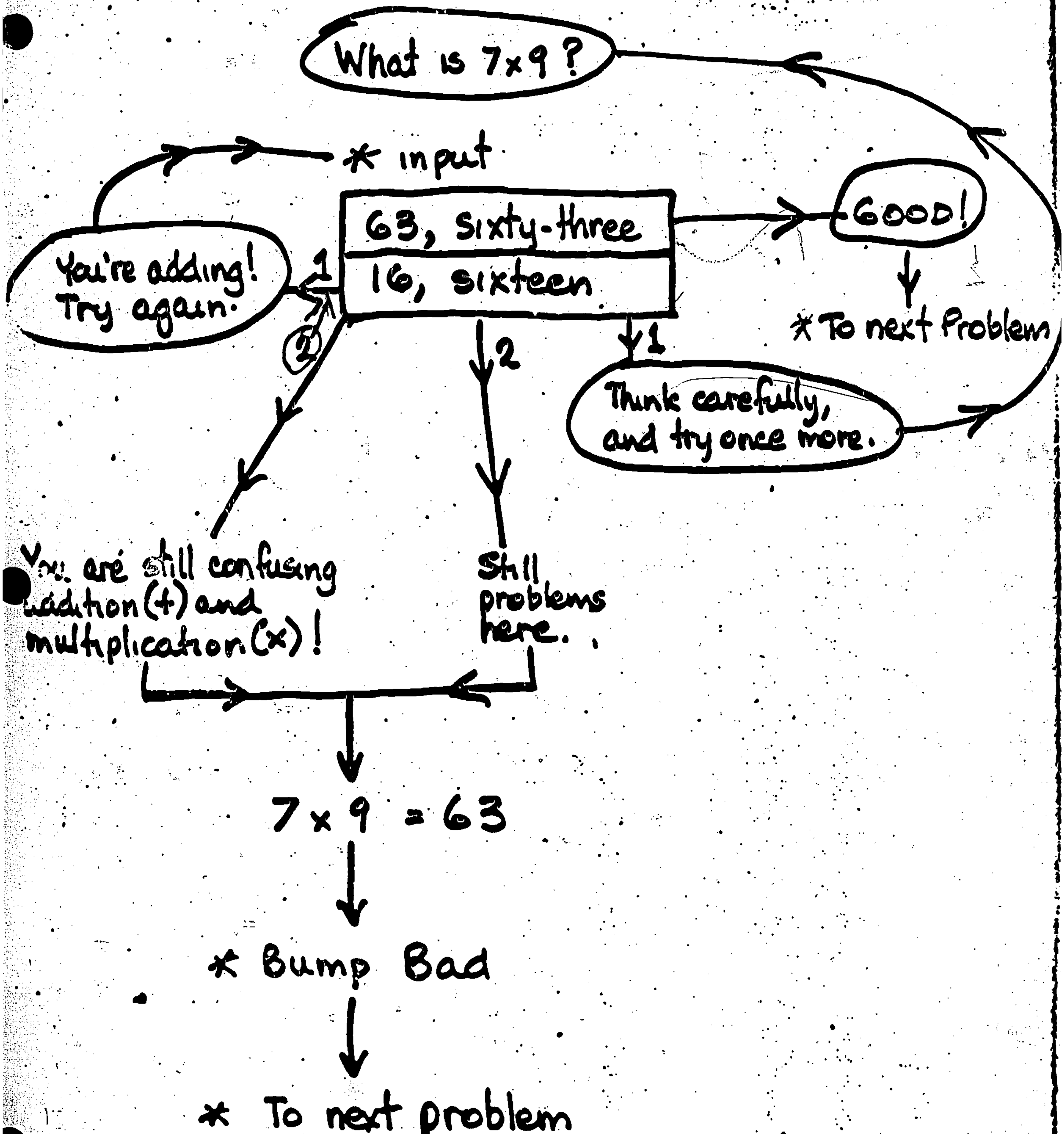
Lipson predicted that the advent of electronic publishing will mandate some new organizations, and the federal government has already addressed this need by funding organizations such as CONDUIT and Data Span. It has also built dissemination components into every contract, published a project book with one-page descriptions of all the projects it supports, developed a support contract with a firm that will fill all requests for information, sponsored a number of meetings, and sponsored publications to help information flow.

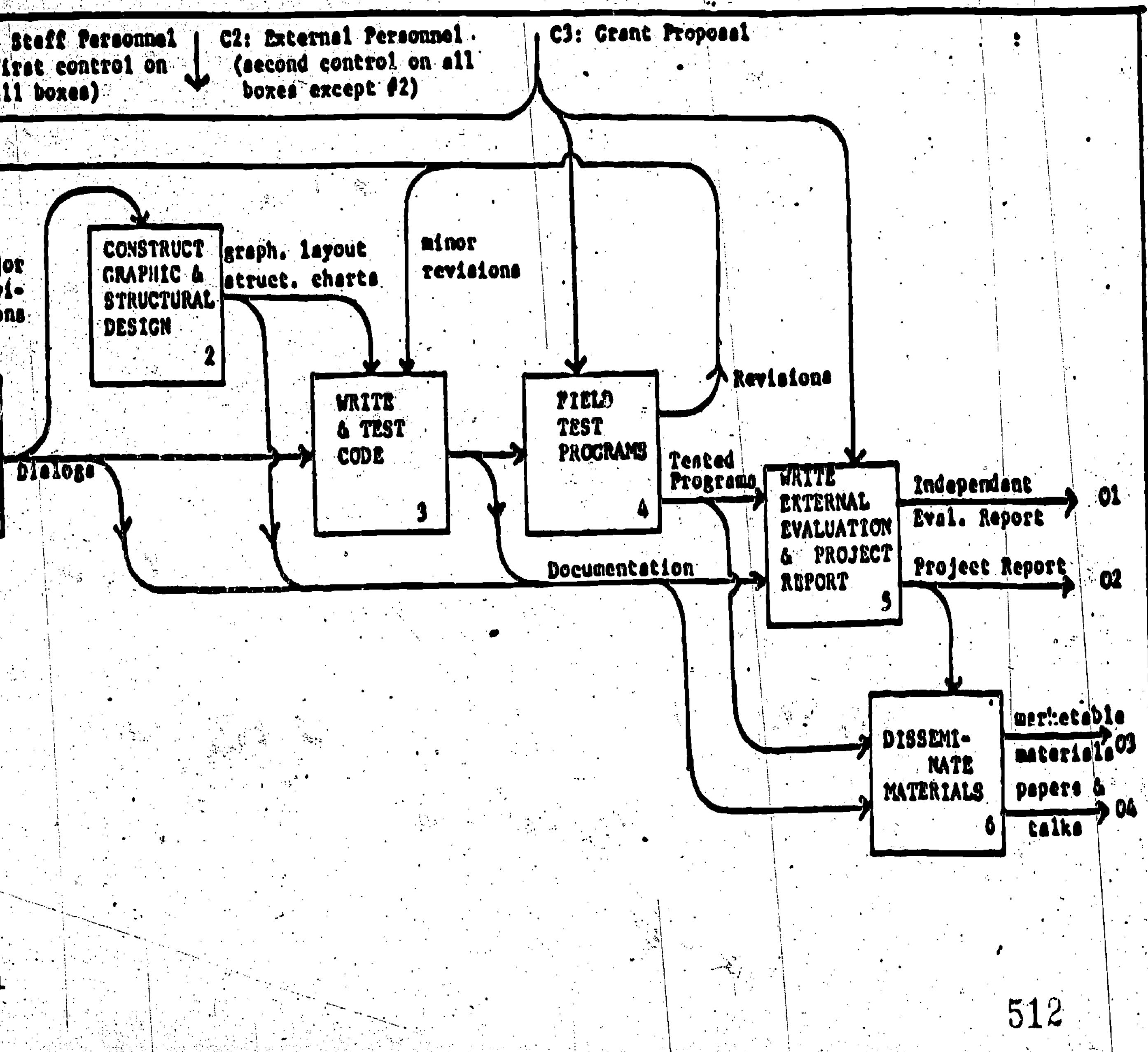
Carl Zinn concluded the session by discussing the Data Span project, which identifies the resource needs of science teachers who use computers in science education (again the emphasis on CAI for science and math). Zinn advocates tutorials on microcomputer use in elementary school. He also supports microcomputers as tools for lecture-demonstrations at the college level, as decision aids, and as aids to teaching authoring on limited microcomputers.

ADDENDUM

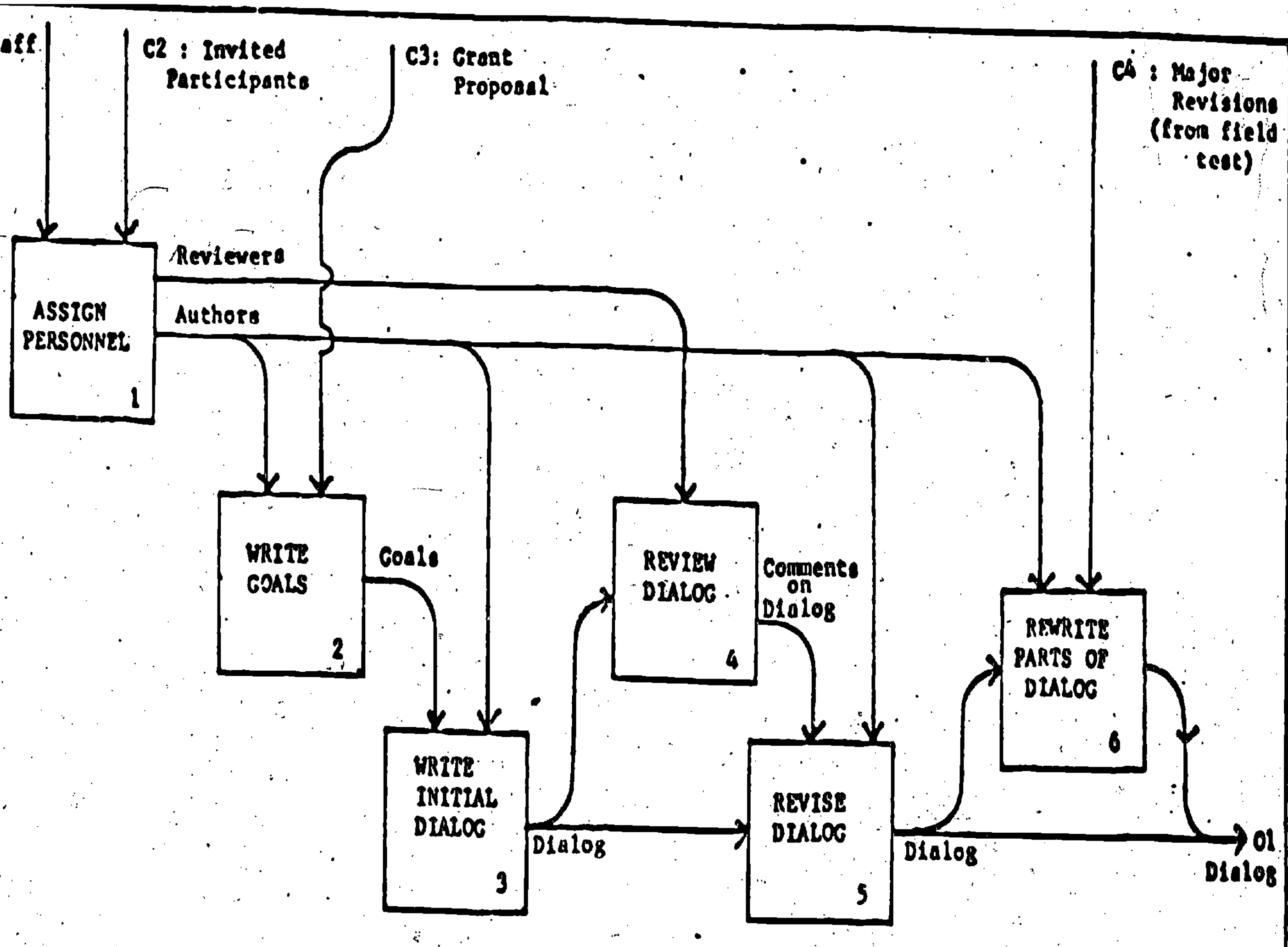
Flow Charts

The Type of Flowchart Sent to
ETC Programmers





Development Cycle for Computer Aided Learning Materials | BLK 1

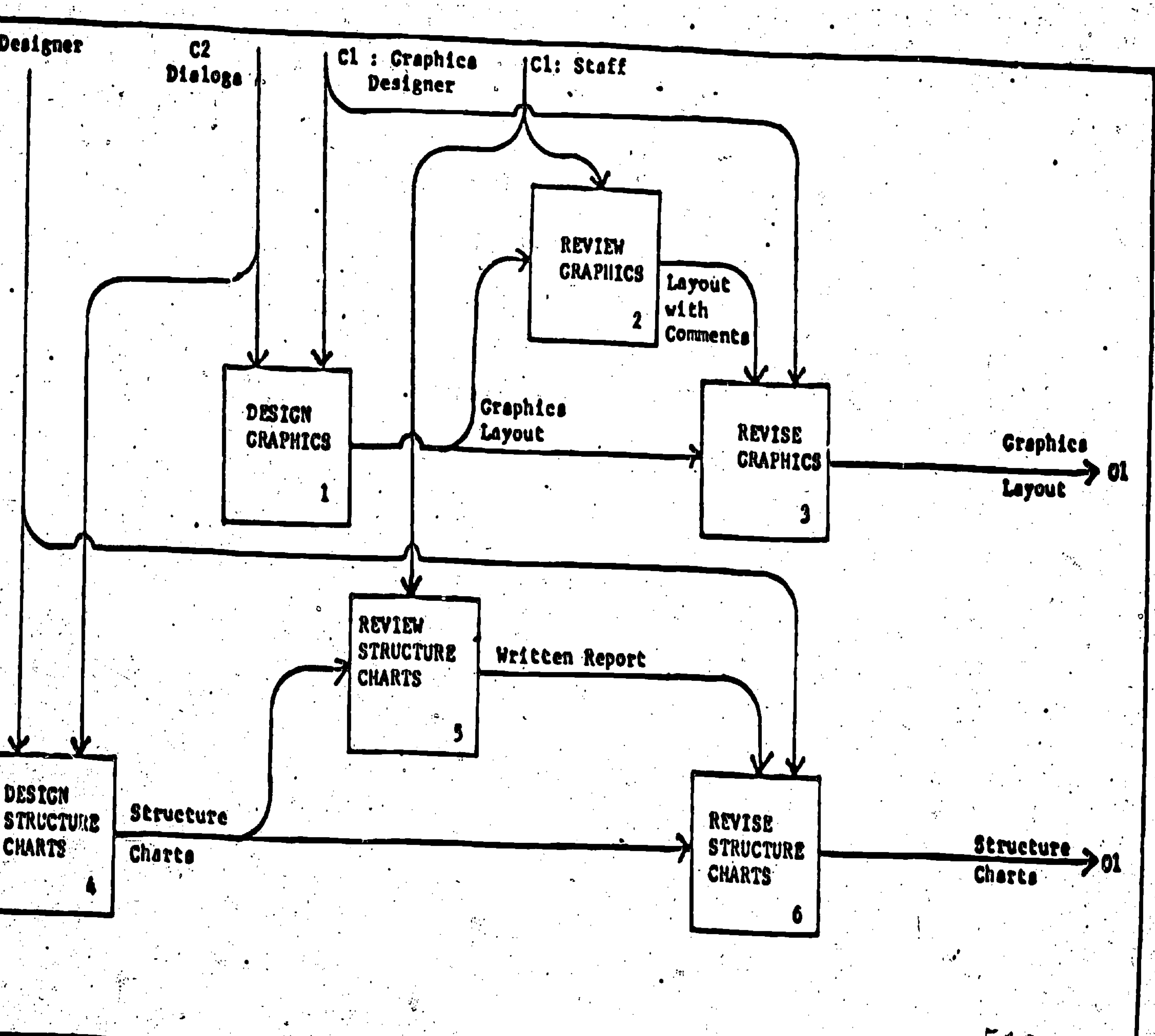


514

Writing Dialogs

BLK 4

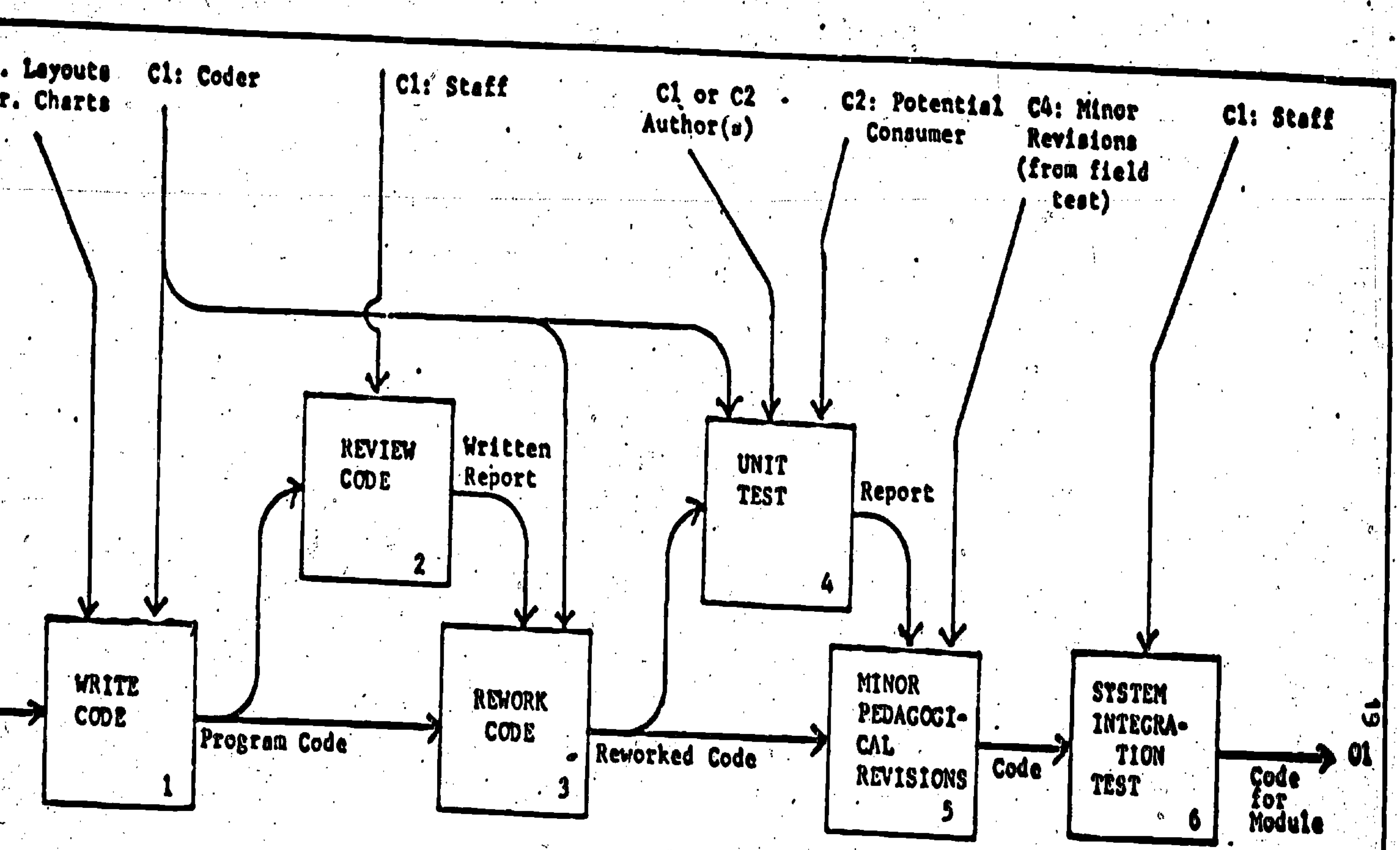
ETC



516

Graphic and Structural Design

BLK 5

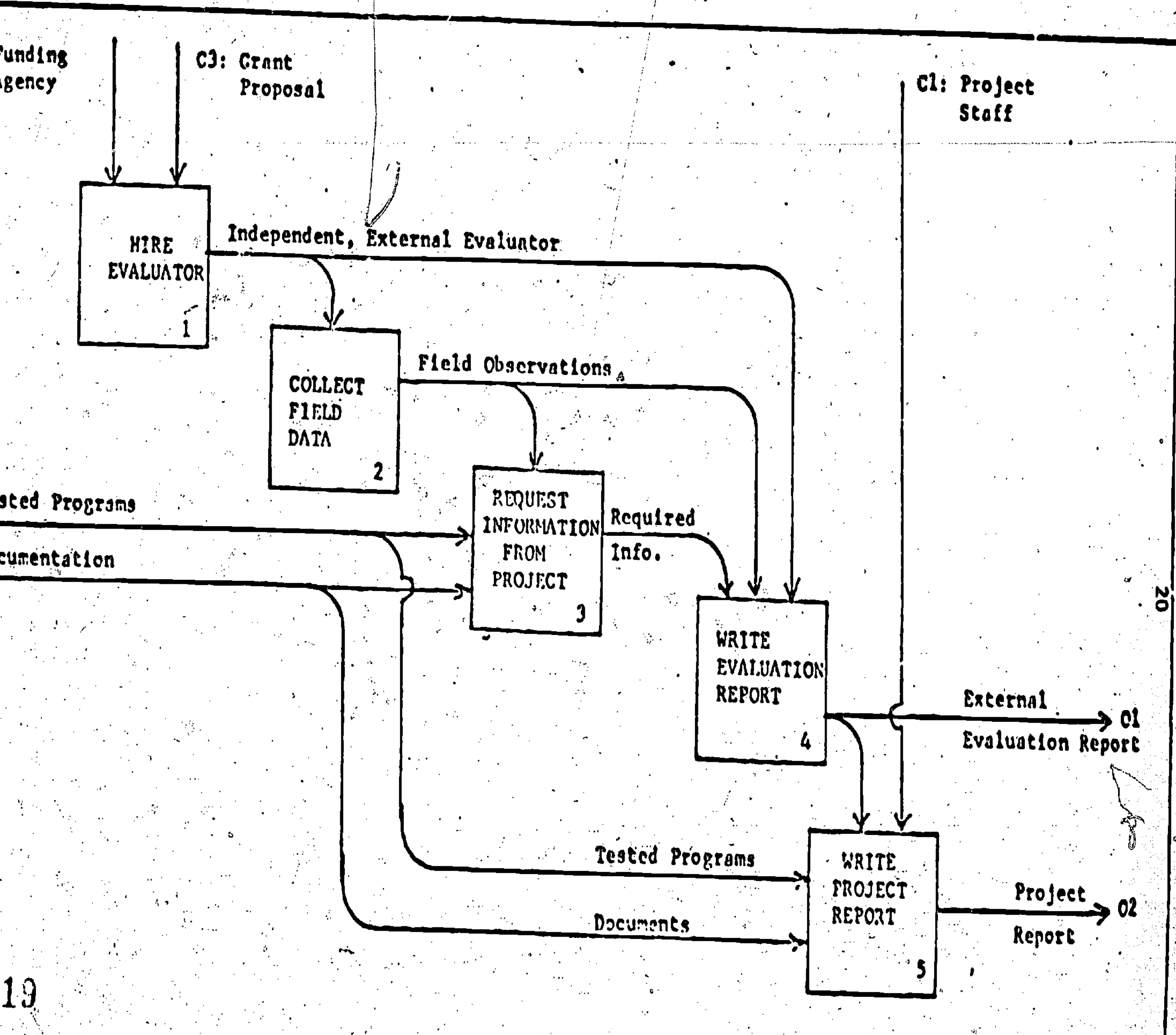


518

Write and Test Code

BLK 6

ETC



19

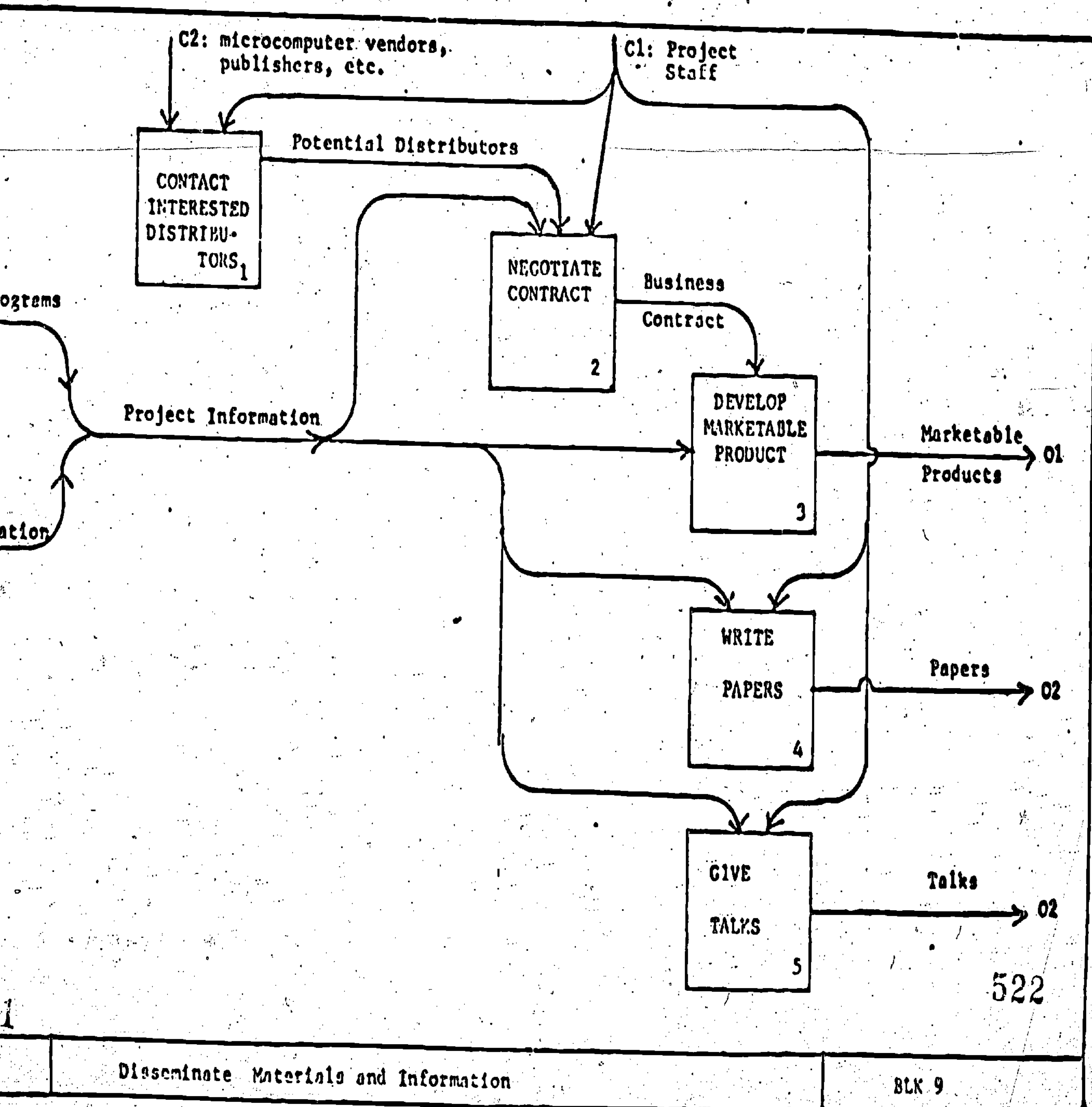
520

20

Write External Evaluation and Project Reports

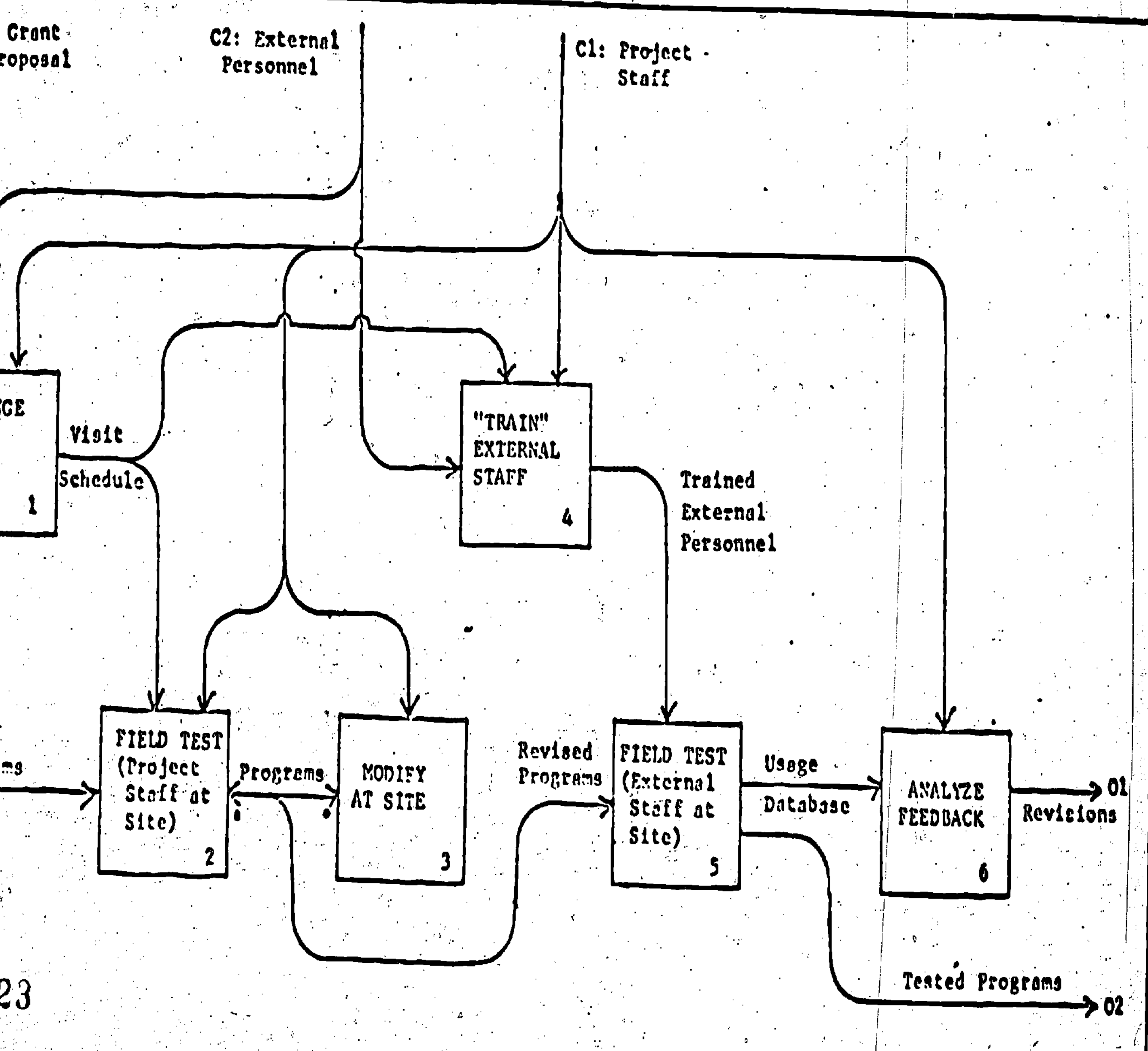
BLK 8

ETr



C2: microcomputer vendors, publishers, etc.
C1: Project Staff
Potential Distributors
CONTACT INTERESTED DISTRIBU- TORS 1
NEGOTIATE CONTRACT 2
Business Contract
DEVELOP MARKETABLE PRODUCT 3
Marketable Products
01
ograms
Project Information
ation
WRITE PAPERS 4
Papers
02
GIVE TALKS 5
Talks
02
Disseminate Materials and Information
BLK 9

522



Field Test Programs | BLK 7

ETC . . . .

MICROCOMPUTER INSTRUCTION AT THE CALIFORNIA READING ASSOCIATION CONFERENCE

ANAHEIM, NOVEMBER 1981

Ann Humes

Attending the California Reading Conference was a reassuring experience. I spent a portion of my day at CRA observing the microcomputer instruction exhibited by various textbook publishers, and I found that textbook publishers, at least those represented at CRA, are far behind SWRL in their approach to microcomputer instruction.

Only one publisher was displaying instruction on a color monitor. However, the text was nearly unreadable: The designers had programmed the material so that the words did not appear in a single color, but in shimmering multi-colors. This material contained the only two graphics in any of the instruction exhibited. One was an instructional graphic that illustrated a container being filled with liquid, and the other was a motivational graphic, a non-animated cartoon character with the word "WOW" printed below it. All the other instruction that I examined was in black and white, had no graphics, entailed much uninterrupted reading, and had undistinguished formatting. In commenting on the instruction displayed at CRA, one of the presenters told the audience that he defied anyone to find any difference between the publishers' textbook materials and their microcomputer instruction.

Microcomputer instruction at CRA was a polar contrast to the flashy, eye-catching materials I saw at the National Educational Computing Conference in June. Yet the visually inviting materials at NECC were not instructionally sound because the developers were not instructional experts. The presenters at CRA had only one suggestion for solving the problem of instruction that does not combine the best of the programming possibilities and the best of instruction: teachers. All the presenters encouraged teachers to develop microcomputer instruction because they have the instructional expertise that computer people lack and the creativity that textbook publishers lack. Yet teachers do not have the knowledge of instructional design and development necessary for producing quality materials. Neither do they have the great amount of time required to refine and debug computer instruction. With teachers, computer companies, and textbook publishers all trying to produce materials for the new instructional medium, the field is likely to become more chaotic than it already is.

The sessions I attended are summarized below; most were interesting and informative.

Steven Marcus, University of California, Santa Barbara, "Computers in the language arts"

Marcus discussed his model of the writing process--prewriting, writing, and rewriting. He asserted that 85% of a writer's time is spent on prewriting, 1% on writing, and 14% on rewriting. He did not offer any documentation for these percentages. He claimed that computers can help in the "writing" stage of the process because of the word-processing capabilities. After explaining these capabilities, he discussed prewriting as important to microcomputer instruction, listing as prewriting activities anything that requires writers to interact with computers. Marcus discussed rewriting as reworking the materials after they are printed in hard-copy form.

The compu-poem program that Marcus is presenting at so many conferences is not very exciting. The program asks the student to type a noun. After the student types in a noun, the computer requests an adjective, then another adjective, then an adverb, another adverb, and finally a verb. After the student has produced all these form-class responses, the computer reformats the words into a "poem." When someone in the audience asked Marcus what happens when the student doesn't type in the correct form-class word, he answered that the computer accepts anything as a noun, etc. He defended his approach by saying that he had been "into computers" for only one and one-half years, so he did not know how to design evaluation into his program.

After the student completes a poem, compu-poem displays a menu of "advice." For example, the student might select from the menu some information on "phrase." When the program branches to "phrase," the monitor displays much text explaining that the student could have used a phrase as the noun for the poem. Marcus also showed slides of several other branches, and all of them involved extensive reading of text and little response from the student. Marcus spent the rest of his time reading some testimonials for his program and reading some of the peoms created by the students.

Marcus had a couple of important comments on writing computer instruction. One was that the teacher should be allowed to turn off any sound programmed into the material. He also commented that the feedback in much of computer instruction is far too sarcastic. It appeals to an adult's rather than a child's sense of humor. Thus it may turn students off to computers. Marcus feels that responses to students should not be negative, but should reinforce or encourage the student to explore further in any problem area.

Marcus ended his session by encouraging teachers to write programs themselves and to send for programs developed by other teachers. These programs are available from Soft Swap at the Santa Clara County Department of Education.

Esther Little and Ronda Winter, Campbell Union High School District, "Turn the computer on to turn kids on to teaching"

Unfortunately, this was the only microcomputer session scheduled for the time period. In Little and Winter's district, a microcomputer laboratory has been organized to help students with their reading. These presenters discussed the background of their computer project and their rationale for it. They presented some slides of their school and of the students in the laboratory at their school. They also discussed how school districts can get money for microcomputers and how students are enthusiastic about microcomputers. Little and Winter also reported that using microcomputers helped remedial readers improve their reading scores.

The presenters discussed the equipment they purchased (Apple) and quoted prices. They passed out a number of handouts, including a glossary of computer terms that they have posted in their laboratory for students to learn.

Richard Tingey, Sequoia Union High School District, and Barbara Tingey, Computer Curriculum Corporation, "Be quiet, computer. It's my turn! You can talk to a computer"

The Tingey's presented background information on computers, saying that computers can be used for instruction, for management of instruction, for keeping records, etc. They then defined the difference between a lesson and a curriculum ("a lesson is a segment of the curriculum"). They talked about computer languages, explaining authoring languages and referencing a new language called "Blocks," which will soon be available from the San Mateo County Office of Education. They also discussed higher-level languages, assembly language, and machine language. The Tingey's then defined some commands used in Basic.

After this general background, Richard Tingey discussed his use of the microcomputer in teaching English. Since much of his time is spent making sure that students have the skills necessary to pass competency tests, he uses the microcomputer to provide worksheets for the students to practice their skills. Tingey writes programs that will generate multiple worksheets. The teachers in the session responded enthusiastically when he said that he can use one program to reorder items, thus creating several "different" versions of the worksheets. He can then replace some of the content within that program and produce still more "different" worksheets. So much for the microcomputer as a device for teaching the composing process!

Forrest L. Miller, San Bernadino Schools, "The microcomputer: A tool for thinking"

Miller is an absolutely delightful speaker, but his presentation had nothing to do with using the microcomputer as a tool for thinking. The purpose of his presentation was to introduce teachers to the uses of microcomputers and to discuss the revolution that microcomputers would effectuate in the schools. Miller listed positive characteristics of microcomputers--they are small, movable, stand alone, are inexpensive, need no special environment, and have graphics and sound capabilities. He briefly recounted a history of computers in education, going back to Eniac in 1946 and Altair in 1975, which was the first commercially available microcomputer. Miller then discussed how computers are used in education, citing CAI, CMI, and CEI (computer enhanced instruction). He also mentioned word processing and data processing for attendance, scheduling, records, etc.

Miller asserted that leadership for the computer revolution must come from the classroom and that these leaders will be people who are new to computers. He lamented that the inadequacy of software was impeding the revolution, charging that currently available software shows little planning and less evaluation. Miller predicted that all of these factors will necessitate a new curriculum.

Miller claimed that computers will be used in education for motivation, for improvement of skills, to provide students with computer literacy, and to save time and money. These factors should be considered in selecting a microcomputer, as well as dealer support, company support, available software and courseware, expandibility, auxiliary storage capacity, number of ports, and quality and number of available languages. According to Miller, some of the biggest problems in microcomputer education are that little long-range planning is done, that the number of text and other materials available are the same as those available in textbooks, and that the user is always waiting for tomorrow's device and for textbook publishers to catch up. He charged that education is just not imaginative enough now for computers.

COMPUTER SESSIONS AT THE CONFERENCE ON COLLEGE COMPOSITION AND COMMUNICATION

WASHINGTON, DC, MARCH 1980

Bruce Cronnell

COMPUTER ASSISTED INSTRUCTION IN WRITING

Edward A. Kline, University of Notre Dame, "Computer-aided review lessons in English grammar and spelling"

Kline has 57 CAI lessons on morphology, syntax, mechanics, phonology (for ESL students), and spelling. Essentially, the lessons seem to be not much more than computerized workbooks. The general principles are presented and the student responds to multiple-choice questions. If the student answers incorrectly, the reason for the error is explained and the student is referred back to the question to answer again. If the student answers correctly, a comment is made on why the response is correct. (This has been added in case a student chooses the correct answer by mistake.) These lessons are used after individual study of a handbook or grammar book. From Kline's description these seem to be very traditional and boring. However, he does note positive teacher and student reactions. Kline said that these programs are available free (except for the cost of tapes) in common computer languages. He is currently converting them to use by microprocessors in Basic using a disc-tape format.

Lorne Kotler and Ron Link, Miami-Dade Community College, "Development and implementation of the RSVP feedback program for individualized analysis of writing"

This isn't really CAI at all; it's a management program. The teacher receives computer sheets on which she or he can check off problems that a student has had in a composition. These computer sheets are fed into the computer and the computer writes a "letter" to the student pointing out these problem areas and giving instruction on the problems. A mass of handouts, with all the information and examples, were distributed.

Margaret Hirtz and Abraham Oseroff, Miami-Dade Community College, "Administration and application of computer-assisted instruction in the teaching of composition"

This is a management system based on placement-test data, age, race, ethnicity, sex, and reading level. Students are assigned specific book pages to study in order to meet the various objectives of the course. This has been implemented for reading but not for writing. Again, extensive handouts were distributed.

DALLAS, MARCH 1981

Bruce Cronnell

COMPUTER ASSISTED INSTRUCTION: VALUES FOR THE 1980's

Mimi Schwartz, Stockton State College (New Jersey), "Computers and the revision process"

Schwartz primarily spoke about the value of using word processors for revision. In fact, she said half of what I was planning to say. A couple of things that she noted seemed to be of interest. One was that some students just can't use computers and/or word processors. (This may be a more serious problem for college students who have had a lot of practice not using computers than for younger students.) The other point was that the clean copies that can be produced by word processors may cover up the fact that poor ideas are being expressed in the writing.

Bruce Cronnell and Ann Humes, SWRL, "Using micro-computers for composition instruction"

Since this paper is readily available, I will not report on it here.

Ken Autrey, Tougaloo College, "The effectiveness of computer writing instruction at Tougaloo College" (handout available)

Autrey reported on computer assisted instruction used at Tougaloo College under a large grant from the National Science Foundation. The programs cover surface features of text and are fairly simple. His studies suggest that students very much like using computers. His handout included sample programs and a bibliography.

Edward A. Kline, University of Notre Dame, "Computer-aided review lessons in rhetoric"

Kline's work continues along the same line as reported last year, except now he is designing lessons to teach rhetorical concepts. All the various bits and pieces of rhetoric are covered by lessons with multiple-choice items. Student answers, whether correct or incorrect, receive feedback from the computer. He noted that all his previous programs for spelling, vocabulary, etc. are currently available for Apple and for TRS 80.

SAN FRANCISCO, MARCH 1982

Ann Humes

BEYOND CAI: IMPLICATIONS OF THE SIMILARITY OF COMPUTER PROGRAMMING AND EXPOSITORY WRITING FOR TEACHING COLLEGE COMPOSITION

Richard E. Walton, University of Montana at Missoula, "Computing's real promise for improving college writing instruction"

Walton described a program at the University of Montana that uses the computer to teach writing. Montana doesn't use CAI: rather, the project introduces computer programming as preparation for writing instruction. Walton asserted that the real promise for computer-aided writing instruction lies in this approach, because writing computer programs and writing expository papers are similar cognitive tasks.

At the University of Montana, three courses comprise the composition program. The first course is a course in programming, the second course is a course in logic, and the third course is a composition course that is based on knowledge learned in the previous two courses. The Montana hypothesis is that programming and writing are similar in objectives and processes, and in the tools required. The one important difference, according to Walton, is that programming is much simpler.

Walton drew these parallels between programming and writing: They are both expressions of understanding; they are both problem-solving tasks; they are both heuristics (problem-solving techniques); both can be understood in a hierarchical order (he is referring here to a top-down emphasis in design).

Gerry Brenner, University of Montana at Missoula, "Programming in writing: The University of Montana cooperative composition experiment"

Brenner restated the argument of his colleague, Walton, that the art of writing programs is like the art of expository writing. He continued by charging that the most difficult problem in implementing the project was the attitudes of English teachers. Getting the administration committed to the project was the other major problem.

For some reason, Brenner digressed to discuss the "unexpected difficulty" in the writing program, which was the method for scoring the writing samples of students who completed the courses. It is interesting that they solved their problem by using a "modified version of an analytic scale." One major modification of the model had raters score for both paragraph-level and for essay-level organization.

The writing-sample scores from the first students out of their experimental program suggested that while students in both the experimental and control groups began the year with a great deal of confidence in their ability to write, at the end of the year the confidence of the experimental group was justified.

Henry R. Harrington, University of Montana at Missoula, "The Wariner-Orr diagram in composition"

Harrington explained how the Wariner-Orr diagram is used for teaching composition because it begins with a top-down programming organization. The first step is to get all the ideas; the second step is to get the major points and describe a logical superstructure. Ideas are listed in four columns of differing levels of content. Harrington claimed that this helps students to look for redundancy. He also asserted that this diagram reveals differences between a thoughtful paragraph and a paragraph of simple elaboration. He added that one of the crucial factors making this diagram a powerful tool is that material can be ordered in any way so long as the relationships remain the same.

Miles Myers, University of California at Berkeley, "Response to the Montana program"

Miles Myers commented on the Montana project by first saying that Christensen is a forebear of this approach to writing. He cautioned that many people in composition prefer to think in terms of pre-writing and in fuzzy sets rather than in the precise, descriptive computer models presented by the Montana people, but their type of teaching does not allow for these dimensions. Myers further responded that teaching must deal first with the question of what exposition is, and that the Montana project deals with a very ritualistic kind of exposition; other dimensions of exposition are not covered by this approach.

OTHER PRESENTATIONS

Shirley Morahan, Northeast Missouri State University in Kirksville, "Designing computer assisted instruction for a writing program"

Shirley Morahan and her colleague Pat Cottie conducted this entire session on computer-assisted instruction. They showed a vic ape of students working on the computer. It turned out that the CAI was on a main frame rather than a micro, but this wasn't clear until the session was well along. It appears that the Missouri people are reinventing the wheel. They did not seem to know much about anything that was going on in the field.

William I. Davisson and Edward A. Kline, University of Notre Dame, "The microcomputer and teaching composition"

This was the session that I chaired. The Notre Dame materials are skills based, and the presentation did not provide any useful new information. Davisson and Kline showed slides of the kinds of instruction they have developed. It is not good material.

SAN FRANCISCO, MARCH 1982

Bruce Cronnell

MICROCOMPUTER CAI: IS IT WORTHWHILE?

G. Michael Barnes, California State University at Northridge, "Evaluating the effectiveness of CAI: Buyer beware?"

Barnes reviewed what is known about computer-assisted instruction. He noted that research suggests that CAI is educationally effective. Studies of CAI have shown either improved achievement in CAI as opposed to traditional instruction or no difference, but no studies have shown that CAI is less effective than traditional instruction. CAI also seems to reduce learning time and to improve student attitude. In addition, CAI seems to be particularly effective with students of lower abilities. He said that these results were consistent across the type of CAI, the level of instruction, the size of the unit of instruction, and the content of instruction. However, he noted that CAI is less effective when it is used to replace the teacher; rather, one must see CAI as a supplement to the teacher. Consequently, costs for CAI must be viewed as additional costs beyond the regular costs of instruction.

Barnes also discussed the actual cost of CAI. He claimed that at present software maintenance is about 50% of the total cost of having CAI in a school, with hardware costs only 10-20% and the rest of the cost being for software purchase. By software maintenance, Barnes referred to the revision of programs to meet the particular needs of students and teachers and to work with different hardware configurations. He said that at the CSUN computer lab (where they have Apple II's), hardware repair and maintenance cost about $2 per hour of instruction. One hour of CAI requires 50-500 hours of development time and, depending on the development time and on the number of students using the instruction, the cost runs from 50¢ to $28.50 per student hour. (He said this information came from an article by Chambers and Specher in Communications of the Association of Computer Machinery, 1980, 23, 332-342.)

Barnes noted that when software is developed, one should be very concerned about portability in order to facilitate hardware replacement and upgrading and to facilitate the sharing of software. He also was concerned about having the source file be accessible to users of programs and suggested that standards are needed for the development of CAI.

William E. Evans, Kansas State University at Manhattan, "Methods and resources in microcomputer assisted instruction"

Evans (who apparently has not done much with computers himself) was concerned with the problems of obtaining and evaluating courseware. He noted that it is very difficult to review courseware before one buys it, although more courseware reviews are becoming available. (He mentioned a magazine called Peelings II, which reviews Apple software.) He suggested eight criteria for reviewing courseware:

1. Does the program run?
2. Is the documentation adequate?
3. Is it easy to use?
4. Is it accurate?
5. Is it educationally sound?
6. Is it adaptable?
7. Does it have appropriate reinforcement?
8. What is the extent of student control over the program?

These all seem like appropriate concerns when evaluating CAI.

Helen J. Schwartz, Oakland University, "A computer program for invention and audience feedback"

Schwartz (who had an article on computers in the February, 1982, issue of College English) described the program that she has developed. The program teaches students to generate information about characters in literature; in other words, this is a heuristic probe program. The program asks the student to name a character, name the piece of literature in which the character appears, and provide a trait about the character (the example was Satan in "Paradise Lost," and the student called Satan tricky). Then the program asks the student to tell what the character does and says that exemplifies the trait, how other people's reactions indicate the trait, how the trait is shown in comparison to other characters in the piece of literature, and (if there is a third-person narrator) how the narrator indicates that the character has that trait. The program includes a "bulletin board" where students can comment on what others have written.

From her use of this program, Schwartz noted that it didn't hurt anybody and that it particularly seemed to help students who were doing especially poorly. It didn't seem to improve good students' scores on essay exams, but their writing in the program did improve.

COMPUTER SESSIONS AT THE ANNUAL CONVENTION OF THE NATIONAL COUNCIL OF TEACHERS OF ENGLISH

BOSTON, NOVEMBER 1981

Joseph Lawlor

SENTENCE-COMBINING ACTIVITIES FOR MICROCOMPUTERS

The two speakers originally scheduled for this session had to cancel at the last minute. Nevertheless, I thought the session was very stimulating and informative. It was also very crowded. I suspect that the NCTE has grossly underestimated the amount of interest in CAI. The three computer sessions on the convention program were all scheduled in tiny rooms, and all were packed to capacity (and beyond).

Robert Morgan, University School, Shaker Heights, Ohio, "Uses and misuses of microcomputers"

Morgan had been scheduled to chair this session, but, with the unexpected cancellations, he had to fill in as a speaker. He first warned against some possible misuses of microcomputers in schools. First, there is the danger that the computer will be used as an expensive arcade game. Morgan suggested that such games might be all right for recreational purposes, but they should never take time away from learning. He also claimed that computers should not become tools only for the "technology freaks" (whether these "freaks" be students or teachers). And finally he warned that computers should not be used simply for their own sake. I think he was getting at something that we have observed in our work: The instruction must drive the technology, not the other way around.

Morgan next outlined some uses for the microcomputer in the English class. He suggested that the machines are ideally suited for the rote work that is an unavoidable part of instruction. Spelling drills and punctuation exercises seem to be less tedious for students and teachers when they are presented via CAI. However, Morgan noted that computers can do much more than drill and practice. They are well suited for teaching concepts, for nurturing creativity, and for stimulating students' interest in reading and language arts. Morgan suggested that sentence combining seems to be a fruitful area for CAI development.

Morgan concluded his presentation by talking about the future of microcomputers in English education. He claimed that the mechanics of writing (grammar, capitalization, punctuation, spelling) will be taught in a much shorter period of time by using CAI, thus allowing more class time for higher-level activities. The development of powerful authoring languages (e.g., PILOT) will lead to a better variety of good courseware. He also suggested that English teachers should learn how to program. According to Morgan, teachers are the ones who know what they need in the classroom, and, through their cooperative efforts, they should be able to meet those needs without relying on commercial publishers.

During the discussion period following Morgan's presentation, several members of the audience suggested that he might have oversimplified the process of developing software. (I think they had a valid point.) Producing software that is instructionally valid and technically sound is a very time-consuming task (as we know). I don't think it is something that can be done out in someone's garage on evenings and weekends. Moreover, I'm not so sure that teachers have the expertise needed to write good instruction (whether it be CAI or paper-and-pencil). In any event, I was glad to see these issues raised.

Irene Thomas, IOTA, "An informal pilot program in CAI for sentence combining"

Thomas apologized for her hastily constructed presentation. She and her husband, Owen, had originally proposed to conduct a hands-on demonstration of their program. However, the NCTE turned down their proposal, presumably because there were "enough" CAI sessions already scheduled. When the original presenters canceled a week before the convention, the NCTE asked the Thomases to fill in. At that point, though, there wasn't enough time to ship the necessary hardware for their demonstration. Consequently, Thomas was only able to show some slides of the computer screen as it ran the sentence-combining lessons.

The Thomases' sentence-combining program has been revised considerably since Ann Humes and I first saw it in October of 1980. Fortunately, they have generally abandoned the multiple-choice formats that were so prevalent in their prototype lessons. However, Thomas noted that students' unfamiliarity with typing is a real problem. (The materials are currently being pilot-tested with fourth- and fifth-graders in the Santa Monica/Malibu Unified School District.) The Thomases have attempted to overcome this problem with a variety of response formats.

For example, in one exercise, the student manipulates an arrow beneath a sentence into which an adjective is to be inserted. When the arrow is in the correct position, the student presses RETURN, and the computer inserts the word into the sentence. In another format, the student constructs the solution to a sentence-combining exercise by selecting words from a list. The screen looked something like this:

Maria knows a woman.

who

The woman draws cartoons.

Which word comes next in the answer: Maria ?

WORD LIST: woman a cartoons knows

draws who

The student moves the box through the word list and selects the next word in sequence. When the student presses RETURN, the selected word disappears from the list and reappears in the answer (that is, if the response is correct). Although these response formats are still rather mechanical, they do seem to be an improvement over the "choose sentence A, B, or C" format that we saw in the original lessons. Thomas also noted that many of the exercises allow students to type in the answer as an option.

I noticed from Thomas's slides that the 64-column display of the TRS-80 computer seems to be an advantage for this type of instruction. The sentences, word lists, and graphic displays fit comfortably on the screen without appearing crowded at all. For some reason, though, the text was all upper case (although I'm almost sure that the TRS-80 Model I has lower-case capabilities). Thomas also mentioned that they are negotiating with Milliken for the publishing rights to their software, which Milliken hopes to adapt for the Apple II.

The Thomases' pilot test began this school year, so they don't expect to report any results until next summer. They have pretested three groups of students: (1) an experimental group, which will use the computer-assisted sentence-combining program; (2) one control group, which will receive pencil-and-paper sentence-combining instruction; and (3) a second control group, which will receive no sentence-combining instruction.

The experimental program contains 20 CAI lessons, each approximately 15 minutes long. If a student finishes a lesson before the time is up, the computer "rewards" the student by allowing him or her to play a word game (e.g., Hangman, Anagrams). The program also includes a management system, which keeps a record of each child's progress.

Overall, I was impressed with what I saw of the Thomases' program. They seem to be far ahead of other efforts in this area. I look forward to hearing more about the project next year.

COMPUTER SESSIONS AT THE ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION

BOSTON, APRIL 1980

Bruce Cronnell

COMPUTER AIDS FOR WRITING AND TEXT DESIGN

Except for one discussant, all presenters were from Bell Laboratories. One of the concerns of the Bell Telephone System is that their technical manuals be readable by their employees. Thus, this is a project that meets very specific needs. However, I think it has a number of implications for our work in writing. In general, the most interesting programs are not yet available; however, they expect them to be available within the relatively near future (after they are patented). A few of the more general text-processing programs (by L. Cherry) are available.

L. T. Frase, "Writer's Workbench: Computer support for components of the writing process"

Frase outlined seven steps in the writing of technical documents: define documentation needs, assemble information, produce draft, assess draft, edit draft, produce document, deliver document. The project has four goals: the computer should 1) analyze text features, 2) compare text with standards, 3) comment on differences from standards, and 4) redesign the text to meet the standards.

Frase felt that this work has instructional implications, and that we must consider the use of microprocessers in instruction since they will become increasingly available.

P. S. Gingrich, "Measures of text coherence"

Gingrich talked about cohesion measurements a la Halliday and Hasan. At present all the computer can measure is use of repeated words.

J. A. Hegarty, "Text reformatting algorithms"

Hegarty is concerned with procedural documents, that is, step-by-step descriptions of how to undertake a task (obviously these are very important to people within the Bell System). She has developed programs that can convert flow charts into lists of numbered steps and from paragraphs to such lists. However, both programs require preliminary human processing.

S. A. Keenan, "Computer projections of the cognitive effects of text changes"

Research has shown that it is generally easier to process texts that are broken down by syntactic/semantic units. Ideally there should be one such unit ("chunk") per line; however, this tends to make sloppy looking text. Keenan's program will derive the optimal line length for a specific text so that, insofar as possible, one line will equal one chunk.

N. H. McDonald, "Pattern matching and language analysis as editing supports"

This was by far the most interesting paper and certainly could be of some help to both writers and researchers in writing.

First of all, there are some simple editing programs. One program will go through and list out all spelling errors by comparing all words in the text to a dictionary. If a writer wishes to use an "incorrect word" (e.g., an abbreviation), it can be entered into a personal list and the computer will treat it as a correctly spelled word. Another program looks for punctuation errors, e.g., only one of a set of quotation marks, one of the set of parentheses, etc. It will identify the incorrect punctuation and show the correct punctuation. In addition, if the writer does not understand why the punctuation was incorrect, information can be obtained from the computer about the specific punctuation rule. A third program looks for words that have been accidently repeated in the text. Another program looks for split infinitives. The fifth editing program looks for commonly occurring awkward or inappropriate words and phrases (e.g., very, different than) that have been derived from style manuals. The computer lists the word or phrase and suggests ways that it might be changed.

Another set of programs looks at style. Some programs will give all kinds of readability measures. Other programs will provide data like number of sentences, sentence length, etc. (to help indicate whether or not there is a variety of sentence lengths). Another program looks at usage of words based on grammatical categories, e.g., number of adverbs, nouns, nominalizations, passives. Moreover, if the writer is interested, he or she may request the computer to list out all cases of these specific items. Another program can compare a specific text with standards, i.e., with these features as found in other kinds of text. For example, it might say, "Your document has sentences that average fifteen words in length while most telephone repair manuals have sentences that average ten words in length."

Other programs available will reformat text in terms of syntactic/semantic chunks and will list out the first and last sentence of each paragraph.

L. Flower, Carnegie-Mellon University, Discussant

Flower was very enthusiastic about such programs. For one thing, she felt that they could do things that people can't or won't do. Moreover, they focus the writer's attention on the text. Such programs can act as an evaluator or monitor, allowing the writer to be a decision maker. Finally, she hoped that perhaps as writers worked with such programs they might internalize some of these editing procedures and use them on their own.

L. Cherry, Discussant

Computers can help writers, especially with tedious parts of the editing process. She cautions that computers are only as smart as humans, and questioned whether what style books say (e.g., mixing sentence length) really makes text better. She cautioned the audience about relying too much on computers since computers aren't always right even with mechanics; people must still be able to edit. She also noted that computers can look at what writers do as they rewrite.

LOS ANGELES, APRIL 1981

Joseph Lawlor

NEW DEVELOPMENTS IN INSTRUCTIONAL SYSTEMS
DESIGN AND COMPUTER-BASED INSTRUCTION: PART 1

Robert M. Gagné and Walter Wager, Florida State University, "Theory-based design of CAI"

Gagné discussed the nine "events of instruction" that must be considered in CAI design; these are discussed in greater detail in Gagné's text, The Conditions of Learning:

1. Gain and control attention.
2. Inform the learner of expected outcomes
3. Stimulate the recall of relevant prerequisite capabilities.
4. Present the stimuli inherent to the learning task.
5. Offer guidance for learning.
6. Provide feedback.
7. Appraise performance.
8. Make provisions for transferability.
9. Insure retention.

Gagné claimed that all of these events need not be present in a single CAI lesson. However, if an event is excluded, then that exclusion should be the result of a conscious decision, and not merely an oversight.

Wager discussed three different types of CAI. First is the drill and practice format, which is intended to provide reinforcement for skills that have already been learned. Second is the simulation lesson, in which the student explores the relationships within a system. Simulations usually require the application of problem-solving skills. The third type of CAI design is the tutorial program, which is intended to be a primary, not a supplemental, instructional system. Tutorial programs can be strictly linear, or they can include branching capabilities.

Alfred Bork, University of California at Irvine, "New developments in microcomputer-based education"

Bork is associated with the Educational Technology Center at UCI. Their work is primarily concerned with (1) designing CAI courses for the introductory science and math classes at the University, (2) designing scientific literacy materials for public libraries, and (3) investigating the effects of screen design on CAI.

Bork identified several issues that must be considered in CAI design: (1) small- vs. large-scale use, (2) present vs. future needs, and (3) single- vs. multiple-screen presentation. Bork also noted that CAI must employ both written text and visual aids. However, he suggested that much of the current CAI software is too text-like: CAI designers must learn to think visually, using more blank space, and they must draw upon the principles of graphic design. Bork also noted that, for long-range production projects, programmers and content specialists should be separated.

Joseph M. Scandura, University of Pennsylvania, "Instructional systems theory: Authoring, diagnosis, and microcomputer-based instruction"

Scandura talked at length about his model of structural learning, which includes the following aspects: (1) identifying objectives via structural analysis, (2) identifying rules that must be learned in order to achieve the objectives, (3) diagnosing individual strengths and weaknesses, and (4) gradually building on strengths until mastery is achieved. Apparently, Scandura has developed a "rule tutor" that can teach any set of rules for performing a task (once those rules have been identified). Scandura's background is in mathematics (as is much of his CAI work), so I'm not sure how applicable his work is to written composition.

Joseph Lipson, National Science Foundation, "Patterns of emphasis on NSF CBI projects"

Lipson discussed some of the science education projects that have been funded through the NSF. He also noted that the NSF will probably face severe budget cutbacks under the new administration.

Lawrence Stolurow, University of Iowa, "Implementation of instructional systems in a microcomputer environment"

Stolurow described the instructional computing programs at Iowa. Apparently, they're doing a lot of things with computers of all sizes (the "microcomputer environment" in the title is somewhat misleading). Stolurow talked about the system itself, with very little reference to specific CAI programs.

NEW DEVELOPMENTS IN INSTRUCTIONAL SYSTEMS DESIGN AND COMPUTER-BASED INSTRUCTION: PART II

This was the second half of the symposium described above. In this part of the symposium, the panel members simply responded to questions from the audience. I've summarized the major points of the discussion below.

A. Problems in the Development of CAI Materials

1. Commercial CAI publishers (like commercial textbook publishers) design their products for a broad national market. However, commercial developers need to become more sensitive to the needs of local users.

2. Commercially developed materials require a long-term program of research and development. Local users, though, don't have time to wait for long-term development. Once they have purchased the hardware, users need software to run on it.

3. Information-sharing networks will become very important in filling the "software void" until commercial developers catch up with the demands of CAI users.

B. Research Agenda for CAI

1. Research is needed to investigate more efficient ways of delivering CAI to students (e.g., computers in the classroom vs. computers in centralized computer "libraries"). More information is also needed on the role of the teacher in CAI. Protocol studies should examine how CAI is actually used in the classroom, and longitudinal studies should investigate the effects of CAI over several years of schooling.

2. Research must examine the effects of CAI on the psychological development of students. Additional research is also needed to investigate how students' progress can be evaluated through computer assistance.

3. More work is needed in designing a sound theory for CAI development. Such a theory can draw from the principles of behavioral psychology, cognitive psychology, and structural learning theory.

4. Research must continue to investigate the efficacy of "intelligent" CAI. (A member of the audience asked for a definition of intelligent CAI, but the panel members couldn't agree on a common definition. As I understand it, intelligent CAI refers to instruction in which the computer makes sophisticated responses to a wide variety of student input. Thus the machine seems to be carrying on an "intelligent" conversation with the student. Apparently, such instruction also includes some very complicated branching capabilities.) Research must also examine how CAI interacts with other instructional media (e.g., videodiscs).

C. Economic Viability of CAI in the 80's

1. The CAI revolution of the 1960's was a flop, primarily because the instruction required expensive "main-frame" computers and elaborate time-sharing networks. Since most school districts couldn't afford these large systems, their interest in CAI quickly faded. In the meantime, commercial publishers had invested a great deal of money in courseware development. Thus, when the school districts backed off, the commercial houses were stuck with expensive products that nobody wanted. Consequently, many commercial publishers are wary of making any large-scale commitments to CAI in the 1980's. If the "micro revolution" is to succeed, it will have to prove itself in the marketplace--without the financial backing of the developers who were "burned" in the 60's.

FUTURE DIRECTIONS FOR RESEARCH AND THEORY IN COMPUTER-BASED INSTRUCTION

Keith A. Hall, Ohio State University, "Influence of microprocessors on CBI research and design"

Hall outlined some of the positive and negative aspects of using microcomputers for instruction and research, and he also speculated on what we are likely to see in the near future:

1. Strong points of microcomputers:

   a. Micro keyboards are much better adapted as input devices than are the keyboards on larger "main frame" systems.

   b. Peripheral equipment for micros is relatively cheap and easy to install.

   c. Micro courseware is readily available at reasonable prices.

d. Research data can be gathered from diverse populations because micros are available in nearly every school district in the country. (I suspect that Hall is overly optimistic on this point.)

2. Drawbacks of microcomputers:

   a. Because of memory-size limitations, storing curriculum and student responses is a problem with micros. Obviously, storing students' responses is critical to research.

   b. At the present time, micros don't allow for communication networks in the same way that larger systems do.

   c. Although micro courseware is readily available, much of it is poorly designed; there is little quality control in the industry.

3. Implications for the future:

   a. Microcomputers will be "clustered" into information-sharing networks.

   b. Much of the CAI research of the 1960's and 70's will be brought to bear on the design of microcomputer instruction. (Hall suggested that many micro courseware designers are reinventing wheels that were developed 10 to 20 years ago.)

Robert D. Tennyson, University of Minnesota, "Research on authoring systems"

Tennyson discussed the Minnesota Courseware Development System. Tennyson does not subscribe to Bork's view that content people and computer programmers should be separated when designing CAI programs. He claims that such a separation is cumbersome: 200 hours of work may be required to produce 1 hour of instruction. Tennyson feels that it is much more productive to have teachers design and write their own CAI lessons. However, most teachers don't have the programming expertise (or the time/inclination to acquire such expertise). Consequently, the Minnesota people designed the Minnesota Authoring System (MAS).

The MAS appears to be a very sophisticated system that allows teachers to control the sequence of their instruction, to design screen formats (using both text and graphics), to evaluate student responses, and to generate student records. Music and speech synthesis are available, and graphics can be entered from a graphics tablet. The computer prompts the lesson writer through a series of menus, and, according to Tennyson, teachers with no programming experience at all are able to design some very sophisticated lessons. (Tennyson showed some transparencies from a sample biology lesson; the text and graphic displays were certainly impressive.)

FUTURE DIRECTIONS FOR COMPUTER-BASED INSTRUCTION: A VARIETY OF PERSPECTIVES

Kenneth Brumbaugh, Minnesota Educational Computing Consortium, "The public school perspective"

Brumbaugh gave a detailed description of the MECC system, which has been in operation for the past six years. MECC uses both timesharing systems and Apple microcomputers to provide an extensive CAI network that reaches every school in the state. Brumbaugh noted that the Apple micros are becoming more popular with the schools, but they are not replacing the timesharing services. There seems to be a role for both large and small systems in the MECC configuration.

MECC has put together a large library of Apple courseware. From what I could tell, they don't have much software available for composition instruction.

Brumbaugh also noted that MECC is expecting some budget cuts in the coming years (as is everybody else in education), but they still expect to keep the system operating. Their long-range planning includes the use of relatively inexpensive micro systems (under $600), which should become available as the technology becomes even more efficient.

Lawrence Stolurow, University of Iowa, "The university perspective"

This was an expanded version of the presentation that Stolurow gave at an earlier session (described above). The Iowa people seem to be doing a lot of micro-based CAI research and development. They have developed software for health sciences, fine arts, linguistics, mathematics, and foreign languages. (Note the conspicuous absence of written composition.) Since 1978, they've also been researching the applications of videodisc technology. In addition, the university provides resources for faculty members who wish to write their own instructional programs.

J. Dexter Fletcher, Defense Advanced Research Projects Agency, "The military perspective"

Fletcher discussed the expanding role of CAI in military training programs. According to Fletcher, the military services are facing a dilemma: Recruits must be trained to operate and maintain increasingly sophisticated equipment, while, at the same time, the educational level of incoming personnel is at an all-time low. The services have found that many of their recruits simply don't have the basic academic skills needed to comprehend written training manuals. Hands-on training has also become impractical because the services can't afford to put multi-million-dollar equipment into the hands of inexperienced operators. Thus CAI seems to offer a solution to the military's dilemma. Through computer simulations, recruits can "walk through" complex procedures, without relying on reading ability. And, if a recruit fouls up in a simulation, there is no loss of equipment (or, I might add, human life).

LOS ANGELES, APRIL 1981

Bruce Cronnell

CURRENT STUDIES IN CAI

Thomas W. Malone, Xerox Palo Alto Research Center, "What makes things fun to learn? A study of an intrinsically motivating instructional computer game"

Malone looked at children's interest in a game-like approach to mathematics because he noted how much children seem to enjoy the various computer games available in shopping malls. He found that boys like fantasy and are not too turned on to verbal responses and that girls perhaps like music with their instruction. He presented some heuristics for designing instructional computer games.

First, the games must have challenge, where there is a goal but the outcome is uncertain. Secondly, they should have fantasy, especially intrinsic fantasy (that is, fantasy related in some fashion to the skill). Fantasy is important because it provides imagery and some emotional appeal (although such appeal may be different for different kinds of students). The third important part of instructional games is their curiosity value. The discussant, Beatrice Farr of the Army Research Institute, pointed out that although this is an important study, we still need to know a lot more about games and instruction; she seemed to suggest the games may in fact get in the way of instruction.

Olivia Sarancho, University of Maryland, "The effects of computer-assisted instruction on Spanish-speaking, migrant children"

Sarancho used CAI with migrant children in grades 3 through 6 to supplement regular reading, language, and math work. This was apparently ordinary traditional drill-and-practice instruction. She found that the CAI students had greater test-score gains than the students not receiving CAI. She also found that the CAI students had a more negative attitude towards the computer than the non-CAI students. She hypothesized that the students are used to the notion of being pulled out of the room for remedial purposes because one is a "bad" student; therefore, they thought they were also being treated negatively when they were pulled out for CAI.

Marjorie Ragosta, Philip Griswold, Paul Holland, Puff Rice, Wang Yu-Chung, ETS, "Does CAI work? An educational experiment"

ETS, with an NIE contract, has done a five-year study in four schools in Los Angeles. It has been a longitudinal study with students in grades 4 through 6. They have done reading, math, and language arts, using already available computer materials. She had a l[illegible] of data, but unfortunately I didn't get a copy of her handout. In general, the CAI worked out well and the report will be available at the end of the year.

Earl Woodruff, Carl Bereiter, OISE: Marlene Scardamalia, York University, "Experiments in computer-assisted composition"

This study was done with sixth graders, although the students had very little time with the computer--apparently only 15 minutes training before they were asked to write a persuasive essay on the word processor. They also wrote a persuasive essay by hand. The students felt they wrote better when they were using the word processor than when they were writing by hand, although this was not true from looking at their compositions.

In the first study, the student could push a "help" button when he or she needed some kind of help. Then the student's composition was erased from the screen and a "help menu" was displayed. There were six things on the help menu:

1. To follow the argument plan

2. To produce the next sentence

3. To tell what you have so far

4. To change words

5. To check unsure words

6. How to quit

Since Woodruff was pressed for time, he didn't explain very much. For number 1, students were given specific explanations of how to write a persuasive essay and were given sentence openers for what they might say in constructing an argument. For number 2, the computer searched for words in the last sentence and asked students to elaborate on them. For number 3, the student's text was displayed from the beginning. For number 4, each word was numbered: Students did not correct individual words using ordinary word processor procedures; rather they decided which word they wanted to change, typed in the number of the word to be changed, typed in the word as they wanted it, and then that change was made in the text. I didn't quite understand how number 5 worked, although students were told not to puzzle over words but rather to put a pound sign after words they were not sure of; my impression is that the computer had some kind of spelling dictionary in it and corrected words, but I don't quite understand how this might have worked. Number 6 was apparently a logging-off procedure.

The second study used eighth graders who had had experience with computers. There were apparently several conditions and I am not sure how they all worked out. In one condition, the computer asked the student a question after each sentence (a sentence was defined as where a period came). It was some kind of question to get the student to go on. This was not nearly as successful as the procedure where the computer asked the student questions after all the writing was complete.

Patricia Gingrich, Lawrence T. Frase, Stacey Keenan, Bell Laboratories, "Computer content analyses in writing instruction"

Gingrich coded two kinds of words into the computer: abstract words and words that had unpleasant connotations. The computer searched through texts to find those that were high and low in abstractness and unpleasantness. Then sophisticated adults were asked to rate these brief passages for abstractness and unpleasantness. The raters agreed with the computer as to abstractness but did not agree in terms of unpleasantness. Gingrich noted that this was because unpleasantness apparently depends a great deal upon context. For instance, although the word kill is generally considered an unpleasant word, in the context of killing insects, most people do not view it as unpleasant.

This study was all part of their Writer's Workbench approach to computer analyses of writing, which they reported upon last year at AERA. Apparently the idea is that the computer would make such searches and then tell a writer, for example, that the text was very abstract or that the text had very unpleasant characteristics to it.

Esther Coke, Bell Laboratories (Murray Hill), "The effect of display context on memory for computer-displayed information"

Coke's study showed that differences in typography (black on white or white on black) can affect memory for content presented on the computer. This is not particularly interesting since I can't imagine many situations where one would want to keep switching the type color. The other variable that affected memory of information presented on the computer screen was the number of screens it was presented on. If two different lists are presented on two different screens rather than on one screen, then the lists are more easily remembered. This doesn't seem to be particularly valuable because I can't imagine why one would want to have lots of screens all around to present lots of lists.

Appendix C

APPENDIX C

BOOK REVIEW

Bork, Alfred. Learning with computers. Bedford, MA: Digital Press, 1981.

--Joseph Lawlor

Learning with Computers is a collection of papers and articles written (or co-authored) by Bork over the past ten years. As the Director of the Educational Technology Center at the University of California, Irvine, Bork has participated in developing dozens of interactive computer programs for teaching science. Consequently, the seven chapters in this 286-page volume contain a tremendous amount of practical information.

Although most of what Bork has to say is applicable to subject areas other than science, some chapters in the book have limited value for those of us interested in composition instruction. Consequently, these chapters are covered only briefly in the summaries below. In addition, much of the material in the book is repeated across chapters (and across papers within chapters). Thus, repeated content is noted, but not summarized, below.

CHAPTER 1: OVERVIEW

Will Computers Replace Books in American Education?

The traditional mode of learning in human history has been interaction with other people. Classicial Greece provided three additional modes: written documents, lectures, and dialogs. With the invention of the printing press, books became the principle vehicles for learning, although almost 200 years elapsed between the appearance of the printing press and the widespread use of textbooks.

Within the next 25 years, computers will revolutionize learning. The computer's ability to provide an individualized, interactive learning environment is its major strength. With the cost of computer technology coming down rapidly, computers are likely to be more important than textbooks in the future.

Computers and the Future of Education

Computer technology is developing more rapidly than any other technological field. However, this rapid technological change has not been matched by an increase in our understanding of the implications of these advances.

In the near future, schools may play a much less important role in education, as computers become more prevalent in homes and in public institutions like libraries. Stand-alone computers will become more popular, especially as they are integrated into multi-media learning systems, which will include videodiscs.

The improvement of the graphics capabilities of computers will lead to new ways of organizing learning--ways that depend more on iconic presentation. Designers of learning materials should explore many different possibilities, including having the student learn programming and having the student interact with pre-programmed materials (e.g., dialogs, on-line tests, learning games, simulations).

The Computer in a Responsive Learning Environment--Let a Thousand Flowers Bloom

This paper is a list of six "theses" concerning computer-based learning:

1) Different students learn in different ways. Responsive courseware must include a variety of materials and techniques.

2) We are only beginning to learn how to use computers in education. Courseware development will likely take years of trial-and-error testing.

3) Valid uses of the computer for learning may depend on the subject matter. "What is highly effective in physics may turn out to be useless for literature." (p. 9)

4) All modes of computer usage (e.g., tutorial, managerial) should be retained.

5) We should not abandon other media (e.g., print, film) as vehicles for promoting learning.

6) The ultimate test of any educational material is whether or not real learning takes place.

The Computer in Teaching--Widely Believed Myths

Myth 1: We must choose between direct and adjunct uses of the computer (i.e., student as programmer vs. student as user of pre-programmed materials). [Both uses may have merit in different disciplines.]

Myth 2: We must have massive equipment to use the computer in edu[illegible]. [Many excellent materials have been produced by schools wit[illegible]nal equipment.]

Myth 3: One computer language is much easier to learn than another. [The way the language is taught is much more important than the language itself; BASIC is no easier to learn than other languages.]

Myth 4: Computers will not change institutional structures. [The computer can provide learning materials at any time and at any individualized pace; consequently, the way schools operate will change drastically.]

Myth 5: Computers are too expensive to be used in education. ["Of all the costs involved in the educational process, computer costs are almost the only ones going down." (p. 11)]

Myth 6: A "CAI language" (e.g., PILOT) will solve our design problems. [The language used in developing courseware is only a minor factor in the total development process.]

Myth 7: Large-scale development projects, such as PLATO, will provide all the answers we need. [Although such projects are useful, they do not exhaust all the design possibilities.]

Myth 8: Valid materials can be developed without teachers' assistance. [Teachers' input is crucial in designing courseware.]

Myth 9: Educational computing requires only minor amount of computer resources. [Sophisticated learning materials can require large amounts of memory and complex input/output facilities.]

(Note that Bork's refutations of Myths 2 and 9 seem to be contradictory. However, he offers no further explanation.)

The Role of Personal Computer Systems in Education

There are two major approaches to using the computer in education. The first involves teaching the student to program the computer. The benefits of this approach are that (1) programming can help develop problem-solving skills, (2) learning to program can help students understand the social impact of computers, and (3) computer literacy is becoming increasingly important in society even for those who do not plan to be computer specialists.

The second approach to using computers in education involves the student interacting with pre-programmed materials. These materials may take one of several different forms: drill and practice, drill and practice with remediation, interactive tutorials (electronic "page turning" does not qualify as a tutorial), on-line testing, testing with remediation, and simulations ("controllable worlds").

As the number of learners and the amount of material to be learned has increased, educational technology has responded with media that can accommodate large numbers of learners (e.g., textbooks, lectures, broadcasts). However, since such media are essentially one-way vehicles, personal interaction between students and teachers has suffered. The computer, though, allows learning to be a personalized, interactive experience.

Stand-alone systems represent the future in educational computing. Such systems have several advantages over large time-sharing systems. Time-sharing programs depend on complex operating systems that are not transportable from one machine to another. Consequently, time-sharing requires substantial hardware investments. In addition, the cost of communicating on a time-sharing system (usually over telephone lines) can be prohibitive. And finally, most time-sharing systems do not have the sophisticated graphics capabilities that are so important in computer-based learning materials.

On the other hand, software for personal computers is much more transportable. Operating systems such as UCSD Pascal allow programs written on one machine to be run on another. In addition, "down time" on a personal computer is not so critical as it is on a time-sharing system because fewer users are affected. Moreover, personal computers can easily be connected to time-sharing networks for functions that are best handled by the larger systems (e.g., accessing large data bases and program libraries).

CHAPTER 2: GRAPHICS

Learning Via Computer Graphics

Learning has traditionally depended on the visual sense. Early humans learned by watching and imitating. With the invention of the printing press, text became the important vehicle for learning. However, pictorial representation has remained important, as is evidenced by the large number of pictures, graphs, and illustrations that are used in textbooks. The most effective uses of graphic design techniques can be found in advertising, and courseware developers can learn much from these techniques.

Computer graphics can assist learning is several different ways. In the sciences, graphs can be valuable devices for illustrating complex concepts. Graphics can also be used to add humorous touches to learning materials. More importantly, graphics can reinforce textual material presented on the screen. Text and graphics can appear gradually over time; they need not appear all at once as they must on the printed page.

Graphics can also be used for student input. Devices such as light pens, graphics tablets, game paddles, and joysticks provide useful alternatives to keyboard input.

Although the term graphics usually refers to non-textual information, text itself also has graphic properties. Computers can take advantage of these properties by displaying special characters, (e.g., pi), subscripts, superscripts, oversized letters, boldface, and underlining. In addition, the computer can format text to aid readability by using short lines and by breaking lines at phrasal boundaries. A combination of right and left justification can also improve readability.

The graphics capabilites of most current computers are still rather crude. Poor screen resolution limits what can be displayed, and dot-matrix characters are hard to read. However, videodiscs should help to overcome the inadequacies of computer-generated graphics.

Standard computer languages do not handle graphics well. The Educational Technology Center (ETC) at Irvine has developed two graphics development systems, one written in Pascal and the other in APL. Both of these systems include a graphics "worksheet," an on-screen editor that allows the programmer to draw various shapes and pictures, access special character fonts, and manipulate the format of screen displays.

Future needs for computer-based graphics design include cheaper devices (which should be forthcoming), color, animation, and simple access to other media (e.g, slides, audio tapes, videodiscs). Developers will have to employ a group approach in designing graphics dialogs, using the talents of subject-area specialists, programmers, and graphic artists. Sophisticated learning materials cannot be produced effectively in a "cottage industry" environment.

APL as a Language for Interactive Computer Graphics

This paper is a technical description of the ETC's APL graphics development system.

CHAPTER 3: EDUCATIONAL TECHNOLOGY CENTER AT IRVINE

The three papers in this chapter describe the history, organization, goals, and funding sources of the ETC. Also included are brief descriptions of the programs currently available through the ETC.

CHAPTER 4: PHYSICS

The six papers in this chapter provide detailed descriptions of some of the programs summarized in Chapter 3. These programs are designed primarily for introductory physics courses. The examples provided in the book indicate that the programs are highly interactive, taking full advantage of the computer's graphics capabilities and calling for a variety of student input. Bork claims that several of these programs, designed for non-science majors, can be adapted to non-school environments such as public libraries and shopping centers.

CHAPTER 5: CLASSROOM

Science Teaching and Computer Languages

Bork compares and contrasts several high-level computer languages and concludes that Pascal and APL are the most appropriate languages for the science student to learn. BASIC is the least desirable language.

Learning to Program for the Science Student

Bork suggests that the science student should learn a programming language within the context of a particular science course (e.g., Introductory Physics). By combining programming instruction with the subject-area material, instructors can provide the proper motivation for learning the language.

Modes of Computer Usage in Science

This paper is another summary of the instructional programs developed by the ETC.

Course Management System for Physics 3 Course at Irvine

This paper provides an overview of the sophisticated computer management system in use at UC Irvine. The system records a tremendous amount of data about student performance on on-line tests and takes care of time-consuming bookkeeping chores.

CHAPTER 6: AUTHORING DIALOGS

Preparing Student-Computer Dialogs--Advice to Teachers

A computer dialog is a "conversation" between a student and a teacher, with the teacher's part of the conversation presented through the medium of a computer program. When planning a computer dialog, the teacher/author should first ask how the computer can have maximum effect on learning in the subject area. There is no point in simply translating standard material from another medium (e.g., a book) to the computer. Such an approach is likely to result in trivial applications. The author should consult with colleagues, who will undoubtedly offer valuable insights. In addition, the author should consider using other media (e.g., film, slides) in conjunction with or in lieu of the computer dialog. The author should also be familiar with the principles of instructional design, although many teachers will be unwilling to take the time to familiarize themselves with these principles.

The next step in preparing a computer dialog is to specify goals. These goals should include both short-term objectives (e.g., the factual information that the student must know) and long-range goals (e.g, the problem-solving skills that the student should develop). These goals can then serve as a reference point throughout the development process.

Several types of computer dialogs may be written. First is the on-line test, which provides immediate feedback and relevant help; such dialogs perform a dual function--testing and teaching. Another type is the remedial dialog. These dialogs present diagnostic questions to the student, branching to appropriate material based upon the student's performance. Such dialogs can generate an infinite set of questions by taking advantage of the computer's random number generator. The interactive proof is another type of computer dialog, in which the student explores a mathematical concept/theory. Another type of dialog is the interactive world. The student supplies parameters for various elements in the computer-based "world," and the machine displays the results of using these parameters. Interactive worlds can assist students in forming hypotheses.

Ideally, dialog authors should work in groups of three. A single author will often miss many opportunities for enhancing the power of the dialog. However, a large group will usually spend too much time debating, thereby reducing the cost-effectiveness of the material.

The development process should begin with a one-page outline of the dialog. Then the details of the program's instructional sequences can be specified. Two approaches are generally used. In the first, authors develop the "mainline" program first (i.e., the program sequence that will be followed by a student who has no difficulty with the material), followed by development of the remedial branches. In the second approach, a "frame-by-frame" process is used. The authors specify, in sequence, all the possible branching that will be generated by various student responses to a single frame. The second approach usually contributes to a better design because it requires the author to keep in mind the responses of a student who is confused. In the first approach, the authors may become impatient about filling in the details of the remedial branches. However, these details are crucial to the success of the dialog: "Dealing with the troubled student, giving full aid, is the most important aspect of the dialog; the very good student does not need much assistance in learning." (p. 181)

The author's relation to the programming language is an important aspect of the development process. Three general strategies are available. In the first, the author uses a CAI language (e.g., PILOT) to write the program. There are several disadvantages to this strategy. First, although CAI languages are purportedly designed for the computer novice, they still require considerable training time to be used effectively. Second, most CAI languages restrict the computer resources that can be used. Finally, since many of these languages are relatively old, they do not conform to modern programming practices.

Another strategy for developing dialogs involves using an "authoring system." These systems typically ask the author to respond to a series of prompts that elicit the information needed to construct the dialog. Although such systems are much easier to use than CAI languages, they severely limit the author's options.

The final strategy available to dialog authors is the one used by the ETC at Irvine. The authors prepare flowcharts of the dialog, including rough sketches of the graphics required. These flowcharts and sketches are turned over to the technical experts, who generate the machine code necessary to run the dialog. This strategy frees the dialog authors from technical considerations, allowing them to concentrate on what they know best--the pedagogy of the subject area. This strategy is also more efficient from a technical standpoint. Secretaries can enter much of the textual material required in the dialog. Graphic designers can format the screen displays, and programmers can write the code needed to implement the logic.

The style of a computer dialog will depend on the individual styles of the authors. However, several general stylistic principles should be considered. The dialog should be sensitive to incorrect student responses, providing the appropriate remediation. The dialog should also be highly interactive, resembling a conversation. Consequently, desirable elements include informal vocabulary, simple sentence structure, and humorous touches (if not overdone). There is some debate over whether or not a computer dialog should use first-person pronouns. Most of the dialogs developed by the ETC use such pronouns, and students have generally supported their use. If the authors of the dialog are identified at the beginning of the program, then the use of first-person pronouns may seem natural.

When evaluating student responses, dialogs should check for part of each desired key word, allowing for common misspellings and mistypings. However, authors should remember that there are limits to how sensitive the evaluation can be. Even the best dialogs will still not be able to analyze approximately 10% of the responses that students give. Consequently, dialog authors should be careful about telling the student that his or her response is unequivocally wrong. For similar reasons, the dialog should never employ abusive language. Dialogs should provide several opportunities for students to answer a given question, but must always provide an exit for the student who does not understand the material. One possibility is for the machine to provide hints and advice after the student has responded with a seemingly incorrect answer.

The dialog must provide feedback to the student, but a variety of messages should be included. Repetitious phrases like "Try again" or "Congratulations" are likely to become ineffective (and tedious). Positive reinforcement for correct responses should occur frequently, but not necessarily after every correct response.

Graphics should be used liberally in dialogs. Techniques like flashing, circling, and coloring can help direct the student's attention. Text should be placed carefully on the screen, "avoiding the tyranny of the left-hand margin." (p. 191)

Feedback from student users provides important help in revising dialogs. However, authors must select in advance the relevant data to be collected; if too much information is collected, authors are likely to get "buried" under the data. One useful technique is to save only those student responses that the program cannot analyze. Another technique is to ask students for their comments after they have completed a dialog. These comments can be stored and examined later by the authors.

Limitations of APL as a Language for Student-Computer Dialogs

Although APL is a useful language for science students to learn, it has severe limitations that make it undesirable for designing computer-based learning materials.

Student-Computer Dialogs without Special-Purpose Languages

General-purpose languages such as Pascal are much better suited to large-scale courseware development than are special-purpose CAI languages.

Large-Scale Production and Distribution of Computer-Aided Learning Modules

Much of the early work in developing computer-based learning technology took place in the 1950's. Development was usually conducted by a single individual or a small group, and distribution was not a major focus of their work. What distribution networks there were were largely informal, as friends shared programs with their colleagues.

CONDUIT was one of the early, formalized distribution networks. CONDUIT originally worked with materials that ran on large time-sharing systems. These materials were evaluated for their pedagogical soundness and for their transportability to other machines. Since transportability was such a problem with time-sharing systems, CONDUIT was usually able to distribute only very simple materials. Recently, the agency has been moving toward personal computer applications, and the transportability problems have become less severe.

In recent years, several factors have made it feasible for larger production and distribution systems to develop. First, the cost of hardware has come down dramatically, and there are estimates that improvements in technology will continue to bring the cost of memory down by approximately 30% annually. In addition, as personal computers are mass-produced to accommodate increased consumer demand, the prices of these units should continue to fall.

In the past, computer-based learning materials were developed and delivered on the same system. However, future development efforts will depend on more expensive and sophisticated systems to develop the materials, which can then be delivered on cheaper systems. At present, adequate delivery systems can be purchased for about $5000. Most home computer systems--in the $1000 price range--are still too crude for serious learning materials, but technological improvements may soon make these systems suitable for delivering sophisticated materials.

As computer-based learning becomes more important in the future of education, which organizations are likely to assume responsibility for large-scale production and distribution? Several possibilities exist, each with distinct advantages and disadvantages.

Hardware manufacturers would seem to be logical choices as production/distribution outlets. Such organizations are already acquainted with marketing software, since software often becomes a major ingredient in selling hardware. Nevertheless, manufacturers have been reluctant to move into the education market. Most manufacturers have had little experience with educational materials, and those who have had such experience may have been disappointed in the past. Moreover, manufacturers may view education as a minor market compared to the business world.

Textbook and audiovisual distributors would also seem to be logical choices. These organizations have had experience producing educational materials, and their marketing systems are already in place. However, many of these distributors may find the new technology unfamiliar or even frightening. Although some distributors are entering the educational computer market, others remain cautious.

Another possibility is the emergence of new companies that will produce and/or distribute computer-based learning materials. A few such organizations already exist (e.g, WICAT), but it remains to be seen whether or not these companies will be a major force in the marketplace.

University organizations may also contribute to the large-scale development of materials. A major advantage of such a development system is that the faculty has greater control over the production process than is typical in a commercial arrangement. However, while universities do handle production well, they are much less efficient in distributing materials.

Retail computer stores may play a role in the large-scale distribution (but not production) of learning materials. However, these stores have traditionally catered to hobbyists, and their marketing strategies are geared toward inexpensive home computer systems and related game software. Consequently, they will probably play only a minor role in distributing sophisticated learning materials. Similar limitations also apply to specialty shops (e.g., bookstores, electronic stores) and department stores.

One interesting organizational structure for producing and distributing materials was suggested by the Carnegie Commission about 10 years ago. The Commission recommended that the federal government establish seven national centers whose sole responsibility would be to produce and disseminate computer-based learning programs. Although the Commission's proposal was based on a timesharing system using mainframe computers, the concept of the national centers would seem to be even more effective with the new microcomputer technology. However, establishing these centers would require massive amounts of funds, and, to date, no action has been taken.

The final type of organization that might contribute to large-scale production and distribution efforts is one modeled after the Open University in England. The Open University is an extension of the correspondence-school concept. Since the University does not have to support a large physical plant, it is able to spend large sums of money on curriculum development. Although the Open University has not made much use of computers, the production process employed by the University provides an excellent model for courseware development. A similar organization could be established in the United States, but it would probably require a great deal of federal financial support.

(The rest of this paper reviews the ETC's system of courseware development. See the summary of the first paper in Chaper 6.)

<u>Single versus Multiple Authorship of Computer-aided Learning Dialogs</u>

A group of two or three authors working together will generally produce better learning material than a single author working alone.

CHAPTER 7: THE FUTURE

The seven papers in this chapter repeat much of the information covered in earlier chapters. Major points are briefly summarized below:

- At present, transporting learning materials across machines is a real problem. In order to develop programs that will run on several different systems, authors usually have to minimize the use of graphics, file-handling capabilities, and sophisticated input/output facilities--the features that vary most widely across computer systems. However, these are also the features that make good materials so effective.

- The UCSD Pascal system holds great promise for helping to overcome transportability problems in personal computers.

- "Intelligent" videodiscs (i.e., videodiscs controlled by computers) have tremendous potential for improving learning materials in the future. However, development must always be driven by pedagogical considerations, not by technology.

- Additional research is needed in two major areas. First, the various theoretical models of how people learn must be tested in empirical studies. Such studies should be media-independent. Second, field research should examine the effects of various media (e.g., textbooks, computers, videotapes, films) on learning.

COMMENTS

Learning with Computers is a comprehensive summary of Bork's philosophy of computer-based learning. His comments on developing and distributing learning materials are important, and the book should probably be required reading for anyone entering the field of courseware development. However, the book also raises several questions.

First, although Bork's system for developing courseware is very sound, it is also very expensive. Producing the kind of sophisticated, interactive programs that Bork recommends requires a major commitment of resources. He suggests that the federal government should provide much of the financial support needed for serious development efforts. However, given the present political climate, such funding is not likely to be available. I suspect that most of the support for developing computer-based materials will come from commercial publishers who are already active in the textbook market. It remains to be seen whether or not these publishers will be willing to finance the kind of development efforts that Bork recommends.

Second, Bork's vision of the future downplays the role of schools and teachers in learning. He suggests that educational institutions will undergo a "revolution" within the next 25 years, as more and more courses are offered by computer, with little or no teacher involvement. However, I am not sure that computers will play so large a role in the future of education as Bork claims. Certainly, computers will have an impact, but I am not convinced that this impact will amount to the revolution that Bork foresees.

Finally, it is not clear how applicable Bork's recommendations are to the field of writing instruction. The interactive dialogs that he describes in the book seem to work very well in the sciences, where course content can be readily specified and student responses can be evaluated with some precision. Writing, though, is not quite so precise, and I suspect that different approaches will be needed in computer-based composition instruction.

Appendix D

Ed Tech Product Reviews

~~feature, and the new User's Guide has been designed and written to allow quick access to complete and unambiguous information.~~

~~What the Hall review fails to suggest is a new problem facing the reviewer, the publisher, and readers. Unlike all other commercially published media, microcomputer software is inherently accessible to rapid revision. This implies that the time between a reviewer's evaluation and the publication of the review must accelerate. Publishers need to revise their current policies if they are to serve and inform their readership.~~

~~Sherwin A. Steffin~~
~~Director of Research and Development~~
~~Edu-Ware Services, Inc.~~

Statistics Software Package. *Computer-based software series of five programs (Applesoft version reviewed). Price: $220 (CP/M version, $350). Published by Robert R. Belanger, Ph.D. (private individual), 541 West Sixth Street, Azusa, California 91702.*

Reviewed by: Ann Humes, Member of the Professional Staff, SWRL Educational Research and Development Laboratory, 4665 Lampson Avenue, Los Alamitos, California 90720.

This set of statistics software is available for two different systems: The version in Applesoft Basic requires 48K RAM, one disk drive, and controller with 16 sector PROMS; the CP/M version runs in compiled MBASIC and requires 64K RAM. A printer with parallel or serial interface is recommended but not required.

The set consists of five disks that perform these statistical procedures, with the indicated capabilities and limitations:

- *Analysis of Variance* (up to ten factors; up to ten levels for one factor, nine or less for the others; equal cell frequencies; only two-way interactions evaluated). Output includes degrees of freedom, mean square, and F ratio for each main and two-way interaction effect.
- *Analysis of Covariance* (up to five factors; up to ten levels for one factor; up to 20 covariates; equal or unequal cell frequencies; only two-way interactions evaluated). Output includes source, degrees of freedom, mean square, and F ratio for each main and interaction effect.
- *Factor Analysis* (various combinations, including, for example, up to 60 variables and ten factors, up to 50 variables and 20 factors, up to 40 variables and 40 factors). Output options include principal components, varimax simple structure, orthogonal-powered vector simple structure, weighted cross-factor simple structure, oblique reference structure, latent root, percent of the variance accounted for by each factor, communalities, oblique factor intercorrelation matrix.
- *Discriminant Function* (up to 100 variables). Output includes discriminant function coefficients, variance of the discriminant function, F ratio, cutoff point, and *z* score cutoff point.
- *Multiple Regression* (up to 100 variables). Output includes multiple correlation coefficient, standard error, regression and residual tables, F ratio, regression coefficient, mean, standard deviation, and intercorrelation.

For both the Analysis of Variance and the Analysis of Covariance, a formula is provided so the user can assure himself or herself that the number of factors and levels, and covariates, if applicable, does not exceed the program limitations described above.

Each disk, or master diskette, contains programs, or sub-routines, used to store and update data, calculate and display results, and support and integrate the data analysis system. A manual accompanies each master. The manual briefly describes the particular statistical procedures on the disk. The manual also provides a tutorial for using the disk, including sample test data and test data analysis outputs.

Procedures

All disks were tried out by individuals with experience in research, design, and development. Disks were examined for their usability with one or two disk drives, and with and without a printer.

Operation

After booting the master diskette, the user configures the program for his or her own system. The program may be reconfigured if the user changes the system (by adding a printer, for example). If the system configuration is satisfactory, the user is presented a menu that allows him or her to create a set of variable diskettes, since data are stored on blank diskettes to allow for an unlimited number of cases. The menu also presents options for adding a new diskette to an existing set of variable diskettes, updating or correcting observations, adding observations, running computational analyses, and quitting the system. The user may proceed directly to any of these options by selecting the corresponding number on the menu. When the user chooses to run a computational analysis, results of the analysis are displayed on the screen or in hard-copy form from the system printer. When the selected operation (other than quitting) is completed, the computer returns to the menu.

Evaluation

Although the documentation attempts to explain the statistical procedures, the user needs some knowledge of statistics in order to understand the explanation; the sample tests should not be used without this knowledge because interpretation errors are possible if certain assumptions or pre-conditions are not properly fulfilled. Furthermore, the manual needs to be refined by a professional editor. This suggestion does not imply that current documentation is not complete and comprehensible; it is. Users had no difficulty operating the programs by following the author's step-by-step directions. However, an editor could improve the readability of the prose, clean up minor mechanical errors, and improve the layout. Then snap judgments about the quality of the product would not be based on its lack of publishing cosmetics.

Each disk loads easily, runs smoothly, and is easy to operate. The operating procedures are quite consistent across all five disks, thus facilitating ease of use. However, some inconsistency in available menu selections across the five statistical procedures was noted, which could initially confuse those using the programs with different menu formats. Specifically, the user enters original data in the Factor Analysis, Discriminant Function, and Multiple Regression programs by making the menu selection for adding data observations; however, the user enters original data in the Analysis of Variance and Analysis of Covariance programs by selecting to go directly to the analysis section.

While calculations are being performed, the screen provides feedback so that the waiting user knows the system is working, that it is not shut down or hung up. The option to discontinue data entry is offered at frequent, appropriate intervals, and for most programs, updating the data can be accomplished by identifying variables to be checked without reviewing all the data.

Users are generally guided through the program with simple-to-follow directions. For example, on the Analysis of Variance disk, the user is requested to enter the number of replications for each cell and then the computer elicits the appropriate data for each replication. The only confusion occurred when the users did not keep records of both the numbers and names of variables.

Two problems with the materials were encountered during the review. The programs could not be configured for two disk drives. However, this was only a minor inconvenience. The other problem occurred when data were being stored on the data diskette. Input was requested when the cursor was near the end of the line, but input at the end of the line caused an error message to be displayed on the next line, followed by another request for input. The previously rejected input would then be accepted.

Conclusion

Despite the two problems described above, which caused inconveniences rather than impossibilities, the five disks comprise a package of good software. They accomplish efficiently and accurately the computing tasks they were intended to perform, within their indicated limitations. Furthermore, this package has been designed and developed at a quality level of software that is very easy to operate, even by non-programming users. □

Update

Since these materials were reviewed (above) the developer has added two new programs and has modified the original programs.

The new programs are "Analysis of Variance (W & B)" and "Matrix Manipulation." The "Analysis of Variance" disk allows both within and between factors analysis. Furthermore, this program may be configured for systems with either one or two disk drives. The "Matrix Manipulation" disk allows for addition, subtraction, multiplication, and inversion of matrices. Unlike the other disks in this set, "Matrix Manipulation" does not require a separate data disk. All the data are manipulated on and output (to screen or printer) from the master diskette. These new programs maintain the good quality of the other disks.

The developer has modified the other programs so that they can be configured for systems with either one or two disk drives. The developer has also added the option for hardcopy printouts for all of the program systems, with the ANOVA programs now permitting the data to be edited and printed before the main computations occur. These modifications make the programs more efficient and flexible for the user. □

The War-and-a-Half Logistics Game. *Computer-based courseware. Price: $15.00. Published by Abt Microcomputer Software, a Division of Abt Associates, Inc., 55 Wheeler Street, Cambridge, Massachusetts 02138.*

Reviewed by: Barbara L. Martin, Assistant Professor, Department of Curriculum and Instruction, Kent State University, Kent, Ohio 44242.

This computer simulation, which requires an Apple II or Apple II Plus with 32K and one disk drive, is a strategy game where the player is responsible for supplying each of ten countries with military forces to maintain "the fragile balance of power in tense regions of the world and to confront enemy forces in areas of unavoidable conflict." The player represents the United States. The object of the game is to keep a balanced ratio between U.S. forces and enemy forces. If war occurs and the ratio tips in favor of one side, the other side loses all its military forces in that theater.

Play begins with a summary of the number of U.S. forces, enemy forces, the hostility status (Cool, Tense, at War), and the travel distance from the U.S. to each of the ten countries. These values change weekly as a result of the player's deployment of U.S. forces, the hostility level, and the distance to be traveled. The rounds of the game are weeks of military activity, and during each week, new conflicts may or may not arise. Game play can be terminated at the beginning of any week.

"Beat the Computer"

"The War-and-a-Half Logistics Game" is basically a "beat the computer" type of simulation. The player makes decisions on updated information without any real context for decision-making; it is a trial-and-error strategy. War and tense hostilities occur without knowledge of how or why they occur. Fortunately, as play continues, the player is able to make some better strategy judgments from previous trials and errors, but actual understanding about how or why to manipulate the variables in particular ways is unclear. The game is user-friendly in the sense that the new information serves as a motivator, but the game provides no prompts or cues to aid in good strategy decision-making.

Educationally, the game is no better (or worse) than other computer the "space world" as a theme for this unit. The user's guide claims these words would be found amusing and interesting. We did not find this to be so. Unusual words in a sentence that lacks punctuation only adds to the confusion. Second, when sample sentences are given to demonstrate a particular rule, a noticeable gap appears in the sentence where the computer will add a comma. Once a student catches on to this, demonstration sentences by the computer may lose their effect. Fortunately, this gap does not appear when students are asked to add commas.

Overall, this program not only deals with a curriculum area that is lacking in courseware materials, but deals with the area in a very professional way. The program is highly recommended. □

School CourseWare Journal (formerly CourseWare Magazine). *Computer-based monthly courseware magazine for microcomputer users. Price: $15.95 per issue. Published by School and Home CourseWare, Inc., 4919 N. Millbrook # 222, Fresno, California 93726.*

Reviewed by: Colin F. MacKinnon, Assistant Professor, Ed Media, Western Oregon State College, Monmouth, Oregon 97361.

This "magazine" provides documented, educational computer programs for pre-college users. Each month one or more short instructional programs are provided the subscriber on a C-10 audiocassette in BASIC. The magazine is available in APPLE II, TRS-80, or PET formats. A teacher's guide and complete program documentation are provided for Business, Consumer Economics, English, Fine Arts, Foreign Language, Industrial Arts, Mathematics, Physical Education, Science, and Social Studies subjects.

The programs provided seem simple and straightforward, load easily, and run without difficulty. Since the cassette loading features of the microcomputer are time-consuming and often difficult for a student to accomplish, it is recommended that *CourseWare* programs, where used, be saved for future use on a disk drive, if one is available.

CourseWare provides the means for an educator to review and own a variety of simple instructional programs on cassette at the low cost of $15.95 per issue. They may be used for classroom instruction *as is* or as programming models for students or teachers learning simple programming in BASIC.

The programs are not intended to be complicated, since the memory constraints prevent it. For this reason, they may not suit more complex instructional needs or programming models.

The magazine, as reviewed, provides an inexpensive cassette source for educational software that is simple in design, well-documented, and easy to utilize in an instructional setting. Like all courseware, however, each program should be thoroughly reviewed by the user to suit each particular curriculum need before it is adopted. Programs can easily be modified by a student or instructor/programmer to suit varying needs. Access to the program itself as a model for teaching programming or developing personalized software seems *CourseWare's* most attractive feature. Limited computer memory utilization and availability on cassette are drawbacks, but not serious ones. □

Editor's Note: The publisher has informed us that this product is available now in diskette format as well as on cassettes.

The Statistics Package. *Computer-based courseware. Price: $179.70. Published by Micro Power & Light Company, 12820 Hillcrest Road, Dallas, Texas 75230.*

Reviewed by: Ann Humes, Member of the Professional Staff, SWRL Educational Research and Development Laboratory, 4665 Lampson Avenue, Los Alamitos, California 90720.

"The Statistics Package" is a set of six lesson disks, each of which can be used independently, covering the principal topics included in introductory statistics courses. The disks are designed to run on an Apple II microcomputer with 32K, Dos 3.2 or 3.3, a single disk unit, and Applesoft in ROM. For some disks, a color monitor is required because differences in graphs are explained by references to the colors of the display. User levels for the first disk are specified as advanced elementary, high school, and adult. User levels for the second and third disks are specified as "high school or beyond." User level is not specified for disks four to six.

Procedures

All disks in "The Statistics Package" were tried out by this reviewer, who is experienced in teaching and in instructional research, design, and development. One or two disks were tried out by each of several other reviewers with backgrounds in development and/or research. Users' knowledge of the subject matter ranged from naive to sophisticated.

Program Operation

The program loads easily, runs smoothly, and is easy to operate. Its operating instructions make use of mnemonics (e.g., press *m* for menu, *r* for review), and users control the pace of the program, reading and responding at their own speed. Users are permitted to skip operating directions if they have used the program previously. To move forward or backward in the instructional sequence, the user calls up the menu.

Graphics and Sound Effects

The instructional graphics are relatively well done, usually reinforcing concepts in instruction or focusing attention on important elements in the instruction. In some cases, however, the displays are no more effective than tables or figures in textbooks. Some graphics need polishing. For example, in one graph the line for zero is submerged into the white area below the numbers and is not visible, yet zero is a legitimate score.

Flashing is used effectively to focus on important elements of text. However, flashing is often overdone: It is not unusual for three different elements to flash simultaneously. This excessive flashing often occurs because words continue to blink after they are no longer the focus of the instructional text, thus detracting from the new focus.

Many graphics intended to be motivational are questionable for an adult audience. These graphics can even have a negative effect; users complained about the tedious waste of time. For example, on one disk the "clever" graphics take nearly 30 seconds to construct the word "Menu." And this time-consuming graphic occurs each time the user calls up the menu.

The sound effects may be motivating to younger students. However, their use is questionable for older students because the sounds have little or no relation to the content. They are merely intrusive. Users found one sound effect particularly annoying—the sounds that simulate typing.

Introduction

The architecture of the instruction is reasonable: Instructional objectives are stated at the beginning of each disk; the instruction is personalized; the instruction is followed by assessment; and the student may review material at nearly any point in the program. However, it is in the implementation of the architecture that the program fails.

The actual instruction is uneven; simple concepts are sometimes belabored, yet difficult and complex concepts are sometimes passed over quickly. Some concepts are presented in long and arduous sections, their difficulty often untempered by important interruptions for practicing the content. Students read frame after frame of text that attempts to explain abstract content in difficult prose style. This style consists of an excessive number of passive constructions and nominalizations and, sometimes, of awkward syntax. The readability level in numerous segments is further sacrificed for "motivating" graphics and sound effects that, as noted above, simulate typing: The text appears on the screen one letter at a time.

While good instruction includes frequent intervals of practice on the content, practice in this program is infrequent, meager, and sometimes inappropriate. Some of the "practice" requires students to input scores of their own choosing. Then the computer does the calculating and table constructing. Entering such data is tedious and unnecessary, particularly when the user must enter as many as 50 numbers, pressing the return key each time. More appropriate practice would have the computer generate simple data sets and let the students do the calculating. This kind of practice is occasionally presented. However, the data are often so complex that calculations must be done off the computer, with either a pencil and paper or a calculator. Another inappropriate practice exercise requires the students to demonstrate by their answers that they can guess what question the computer is asking.

Sometimes the only practice is an optional true/false quiz, and many of the quizzes consist of only two items. Furthermore, quiz items are usually either too easy or too silly and inappropriate, as is evidenced by these true/false items:

> The common name for a normal curve is a ding-a-ling curve.
>
> Only a fortune teller can make accurate predictions of the future.

When students answer items correctly, the computer reinforces them with one of several appropriate responses. When students answer items incorrectly, the computer merely informs them that they are wrong and should try again. No explanation is given; no branching to additional instruction is provided. On some disks, students must try again endlessly. On others, the correct answer is presented after the student errs several times. Occasionally, the computer responds with an insult: "You've got to be kidding" was the response to a close guess; "A good choice for someone like you" was the response to the user's choice of an easy rather than a difficult quiz.

To the program's credit, students may review at nearly any time during the lesson. However, the review does not provide new ways of looking at the subject matter; the program simply presents the same instructional text, with minor changes in the graphs or in the data.

Errors

The disks have a few programming errors. Disks three, four, five, and six do not personalize the instruction if the user skips the preliminary operating instructions. On three of these disks, the user is called "Ed," and on one disk, "Student." Another programming problem traps users in a loop at the end of disk six.

Numerous textual errors were noted. The prose is replete with punctuation errors of various types; it has a few minor grammatical errors. Numbers on a couple of graphs are asymmetrical. One table displayed a frequency of zero for a score interval of 67-73, yet the highest score obtained was 72.

Conclusion

"The Statistics Package" is poor courseware primarily because it presents complex content with few opportunities for interaction, and that interaction is often questionable or trivial. By failing to involve the student interactively, the program becomes little more than an electronic textbook. In fact, traditional textbooks have an advantage over this program in that students can move back and forth freely among their pages, studying procedures and examples, and juxtaposing practice items with those procedures and examples in order to understand the content.

It may be unfair to fault this program for problems that are common to much courseware. However, these problems must be addressed so that the quality of computer instruction improves. □

Response by Publisher

"The Statistics Package" has been and is a very effective supplementary teaching tool. It has proven especially helpful to students needing more time to study, review, and practice the basics, at their own speed.

The first three program disks have been used extensively in high schools, The entire set of six disks is now used in a number of colleges and universities across the nation.

For these reasons we felt it necessary to reply to the more specific points and concerns mentioned in the review:

CONCERN: On one disk the program takes too much time to construct the menu screen. REPLY: This is being corrected.

CONCERN: The occasional typing sound-effect is distracting. REPLY: Some find such occasional sound effects distracting, while others welcome the familiarity.

CONCERN: Practice opportunities are "... meager, and sometimes inappropriate." REPLY: At most points students may work the practice exercises indefinitely, until they feel comfortable with the material. The author considers the practice materials appropriate, based on his years of classroom experience.

CONCERN: At one point the entering of score data must necessarily be tedious—having to enter 50 scores. REPLY: This is incorrect. The program simply invites the student to enter "as many as" 50, but will accept but a few.

CONCERN: The occasional true/false questions posed are poor practice and quiz questions. REPLY: These are not intended to serve as either practice or quiz questions. They are merely meant to keep the student attentive, while simultaneously affording some light-hearted relief!

CONCERN: On occasion, students cannot progress in the program until they enter the correct answer. REPLY: We are not aware of this problem. Throughout, students may request review or return to the menu to select the next topic.

CONCERN: Light-hearted program responses are often paired with similar student entries. REPLY: The programs are now being reviewed to ensure that no such responses may accidentally be insulting.

CONCERN: The program does not always personalize its responses with the correct name. REPLY: This may occur when the normal disk start-up procedure is bypassed—a common practice when reviewing software, but not a practice followed by actual student users.

CONCERN: An apparent loop problem exists at the end of disk six. REPLY: We are unaware of any such problem but are investigating that possibility.

CONCERN: The disks contain errors in punctuation and grammar. REPLY: As such errors are brought to our attention, their correction is scheduled for inclusion in subsequent program releases.

For the most part, the concerns expressed in the review are concerns of style and technique. The more essential points—that the program presents basic statistical concepts and techniques accurately, with good illustrations, and ample practice opportunities—seem overlooked.

Instructional strategies and teaching techniques vary. Style, a very subjective matter, plays a major role. We intend to honor and keep the author's style intact, while continually working to correct errors and weaknesses—as they may be discovered and brought to our attention.

E.E. Frantz
President
Micro Power & Light Co.
12820 Hillcrest Road
Suite 224
Dallas, TX 25230

Touch Typing Tutor. Computer-based courseware. Price: $25.95. Published by Xtra-Soft, Inc., P.O. Box 91063, Louisville, Kentucky 40291.

Reviewed by: Frederick G. Knirk, Professor, Department of Instructional Technology, University of Southern California, Los Angeles, California 90007.

"Touch Typing Tutor" has excellent graphics and generally well-designed instructional drill-and-practice exercises. The program progressively leads the learner from home position exercises to coordination and finger exercises. Intermediate testing helps the learner/teacher to spot problem areas.

A Heathkit or Zenith 89, or an H-8 microcomputer with an H-19 terminal having an HDOS or CP/M disk operating system with one disk drive is necessary to run this program.

The first lesson is too cute and time-consuming—it can be skipped. The typing illustration bored the learners who participated in this review. The program can be used by almost anyone. The vocabulary in the first lesson (proficiently, diligence) somewhat reduces its relevance to elementary school users.

The program has a timed practice lesson, which immediately calculates the learner's typing speed.

The content of the program is correct and logically presented.

While no objectives are included with the package, the purpose of the package is clear: to help an individual learn how to type. Because of the organized content, the interest or motivation by the learner should be relatively high.

The documentation accompanying the program is outstanding. The keyboard illustration included with the documentation is a useful learning aid. This CAI program works well, and I rate it as very good. The graphics are excellent; the relatively high resolution of this computer system is well used in this program. The intended users easily operated the program.

This program is logically taught by a computer because most computers depend upon knowledge of operator typing ability. The computer also provides feedback to the user regarding his or her progress and ability to "hit" the right keys, and the intended users should be able to operate the program with ease in normal usage, as did the students who participated in this review. □

Any Disagreements?
Readers who disagree with comments made by *Ed Tech* reviewers, based on their own usage of products reviewed in these pages, may submit Letters to the Editor for possible publication.

Appendix E

SOUTHWEST REGIONAL LABORATORY TECHNICAL NOTE

DATE: August 21, 1980

NO: TN 2-80/23

MICROCOMPUTER SYSTEMS: A TOOL FOR INSTRUCTION

William Russell

ABSTRACT

This report reviews the state of the art of microcomputer technology with emphasis on its potentials and limitations for instructional applications. It also deals with specific issues of importance to the SWRL microcomputer-based composition instruction project.

MICROCOMPUTER SYSTEMS: A TOOL FOR INSTRUCTION

SECTION I. INTRODUCTION

This report reviews the state of the art of microcomputer technology with emphasis on its potentials and limitations for instructional applications. It also deals with specific issues of importance to the SWRL microcomputer-based composition instruction project. (No familiarity with computer systems is assumed, but the technical background introduced in Section II is necessary for some of the ensuing discussion. A computer specialist will find some of the technical background in that section so simplified as to be uninformative, and can skip Section II without loss).

The microcomputer came into being as by-product of larger computer systems. Larger computers are typically operated from terminals that have a typewriter-like keyboard for input and either a screen or a printer for output. Large computers can support a large number of such terminals doing different tasks simultaneously. These terminals usually have at least a small amount of data storage capacity to buffer the ebb and flow of information. More recently (with the advent of the microprocessor chip) they have been equipped to do some smaller information processing tasks themselves to avoid unnecessary use of the large computer. When processing features are added to a "dumb" terminal, it becomes a "smart" terminal, which is little different from a stand-alone microcomputer. In fact, it does history little violence to say that microcomputers were born when smart terminals began to be marketed as stand-alone computers.

In the few years since microcomputers have been widely marketed, they have become complete systems with a large line of supporting hardware and software. In fact the line between these smaller systems and the larger systems that spawned them has been blurred. Some microcomputer systems can now even support networks of "dumb" terminals.

Microcomputer systems have very recently attracted the attention of educators as an alternative to the larger computers that were, until recently, the exclusive means of delivering computerized instruction. The main attractions of microcomputers are low cost, portability, and complete classroom control of operations and scheduling. The advent of the microcomputer appears to have restimulated interest in "computer-assisted instruction." The present report looks at some of the characteristics of microcomputer systems that have sparked this renewed interest.

Throughout the educational community, there is widespread evidence of the new interest in micocomputers. CONDUIT, a clearinghouse of primarily college level computerized instruction headquartered at the University of Iowa, has recently begun to disseminate programs for microcomputers; Northwest Regional Education Laboratory in Oregon recently established MicroSIFT, a clearinghouse for microcomputer programs and information for elementary and secondary schools; and locally information from only two vendors indicates that at least fifteen districts purchased microcomputers during the past few months.

An example of renewed interest in computers for classroom use is the increasing popularity of the Apple microcomputer. The Apple has been included by the states of Minnesota and North Carolina in their statewide computerized education services, and by the Province of British Columbia in Canada. It has also been chosen for computerized instruction research and development activities of Lawrence Hall of Science at the University of California in Berkely, California. Bell and Howell has begun marketing the Apple computer under its own name. Six-hundred Apples were recently ordered from Bell and Howell by a computer consortium centered at the Region IV service center in Houston, Texas, which serves about one-third of the population of that state. Bell and Howell supplies materials with its computers that enable teachers to author their own computerized instruction. The Apple Computer Company itself will soon offer a special programming language, PILOT, which can be used to author computer assisted instruction.

In spite of this renewed interest, a number of critical problems stand in way of widespread, high quality computerized instruction. Probably the greatest of these is the lack of compatibility among competing microcomputer systems. As a result, most instruction for use with microcomputers is prepared by teachers for their own use or by other individuals who do not have the large scale resources necessary to create systematic instructional programs. The limitations of microcomputer systems for instruction are discussed in Section III of this report, but first, in Section II, we establish the necessary technical background for discussing microcomputer systems.

SECTION II. A TYPICAL MICROCOMPUTER SYSTEM

One service the present report attempts to perform is to provide educators with a brief description of a typical microcomputer system. The present section of this report provides such a description. The system is described here with sufficient detail to support discussions of it in the balance of the report, but without mention of variants and qualifications that would unnecessarily complicate these discussions. The reader who has limited knowledge of computers may need to return to this section from time to time to clarify the role of a particular component in the system.

In this section, we distinguish between "hardware" and "software." By "hardware" we mean the set of electronic and mechanical devices that are linked together to make up a microcomputer system and by "software" we mean the intangible set of magnetic and electronic impulses that are programmed into the hardware and stored there to guide the hardware in performing useful functions. The software shapes and schedules all of the operations of the system and the hardware executes them.

HARDWARE

A typical microcomputer system is made up of seven more or less discrete hardware units (components) plus a set of systems software that integrates the functions of the hardware components and provides a practical way for applications software and human users to interact with the system. The seven hardware components are:

1. A central processing unit (CPU) including internal data storage space (internal memory). We include the most rapidly accessible types of memory, RAM and ROM, in the CPU where some descriptions would not. RAM stands for "random access memory" and ROM means "read only memory." This is explained shortly.
2. A power supply
3. An interface bus and possibly other interface hardware that relates the CPU to input and output devices
4. A keyboard
5. A screen
6. A system for using external storage (external memory). By "external storage" we mean storage typically managed by a hardware component separate from the CPU. This is sometimes called "mass storage."
7. A printer

Listed below is some information about each of these hardware components. This is followed by information about the systems software

<u>The CPU</u>

This is the heart of any system. It processes numerical and numerically represented alphabetic data. The crucial piece of hardware within the CPU is a microprocessor "chip," and the type of chip sometimes determines whether the system can run a particular applications program.

Two families of chips commonly used in microcomputer systems are represented by the 6502 chip used in Apple and Pet computers and by the 8080 and Z-80 chips used in the TRS-80 and in many business microcomputers.

Two types of rapidly accessible storage (memory), RAM and ROM, are associated with the CPU component. The full designation of RAM, "random access memory", is not very descriptive of its actual function. RAM provides temporary storage space for software and working space for processing. ROM (read only memory) is typically fully programmed at the factory and cannot be added to. It is used for primarily for systems software, which we discuss later below.

A stricter definition of the CPU might not include these memory components, but for our purposes RAM and ROM should be thought of as "internal" storage, intimately associated with the CPU. Later below we distinguish "external" storage from "internal" storage. By our definition, "external" storage is typically an external physical component while internal storage is physically a part of the component that houses the CPU.

Various peripheral devices allow data to be entered into RAM (internal) storage. These data are processed by the CPU while in RAM and the size of the RAM in kilobytes (K) sets limits on how much data can be processed at one time. ROM storage comes from the manufacturer with information in it that helps the system to operate, and no new data is normally input to ROM once it has been programmed, although one version of this type of storage, programmable ROM or PROM, can be programmed by the user and then left resident in the CPU like other ROM. That is to say, the major difference between RAM and ROM storage is that information stored in ROM is a permanent feature of the system while information in RAM storage is lost whenever is replaced by new data or when the system is turned off. To save data from RAM, it is necessary to deliver it to some type of external storage or to print it.

The Power Supply

The power supply converts wall current into the proper type of power for the system. A single power supply contained in the same box as the CPU usually delivers enough power to support all of the components of the system. When extra components are planned, it is important to know if this power supply will support them.

The Interface System

This may be called a "bus." In any case, it determines what input and output components are compatible with the CPU. Interface hardware can often be added to the system to make it compatible with more devices, such as particular printers or external data storage devices.

The Keyboard

This is the principal input device for most microcomputer systems used for education and small-business applications. It may be something like a calculator pad or it may be more like a typewriter possibly with a pad as well.

The Screen

This and the printer are the two most common display devices. The screen may be color (for graphics) or monochrome. It is typically from nine to twelve inches across, measured diagonally. Interfaces are available for using a regular television set.

The External Storage System

External storage space can be much larger than internal RAM and ROM storage. Also, it is permanent, whereas RAM is not, i.e., information in RAM is lost when the system is turned off, but information in external storage is filed there for as long as the user needs it. The main disadvantage of external storage over RAM is that access to it can be much slower than access to RAM. Some of the kinds of information that an educational user may want to keep in external storage are systems software, pupil records, indexes and dictionaries of information, text such as tests, worksheets, report forms, and applications programs that will enable the system to manipulate these and other data.

As we noted under the description of the CPU above, we are considering RAM and ROM storage to be "internal" and storage which is typically located outside the CPU component to be "external". Another name for what we are calling external storage is "mass" storage. Types of external storage media for small systems are:

1. Cassette tape
2. Floppy disks
3. Hard disks

Cassette tape is inexpensive but very slow. It is necessary to wind serially along the tape to look for a particular piece of data, and many minutes are added to some frequent applications by this process. This is the external storage medium for many small systems in classrooms today, but it is assumed throughout this report that cassette storage is inadequate for most serious educational applications.

Floppy disks and hard disks are similar to phonograph records. They are "read" from and "written" to by means of a disk drive, which is something like a small record player in an enclosed box. The disk is inserted manually into the drive and inside there is an arm that reaches out rapidly to an exact data location while the disk is spinning, whenever data is needed.

Floppy disk systems are much less expensive than hard disks--hundreds compared to thousands of dollars. Smaller size (5 1/4 inch) floppies store from 90 thousand to 300 thousand alpha/numeric characters each, and most applications can access the information they need from this type of storage without inconvenient delay. On the other hand, hard disk systems are available for about four thousand dollars that will store over ten million characters and access this information even more rapidly than floppy disk systems. For a price, much larger hard disk systems are available for microcomputer systems. This is made more attractive by the fact that a practical means has recently been developed that allows up to sixty-four microcomputer systems to share a single large hard disk, provided they are within a one thousand foot radius. One such system is manufactured by Nestar Corporation in Palo Alto, California.

The Printer

There are two types of printer, differing in cost, speed, and print quality. The first type is a dot-matrix printer, which is fast (e.g., 80 to 150 or more characters per second), and can be less expensive than the other type ($750.00 to $3,500.00 or more), but delivers only moderate quality print. The other type is a letter quality printer, which is usually much slower (15 to 45 characters per second), relatively expensive ($3000.00 or more), but delivers high quality print.

In many applications, one printer can be shared by several microcomputer systems. This is much more simply accomplished than sharing
disk storage.

THE SYSTEMS SOFTWARE

Software supplied by the manufacturer provides the system with operations for controlling the external data-storage component, translating applications programs into machine language (the impulses understood by the CPU), interacting with peripheral components (some of which may have their own systems software), and generally running the system.

A "monitor" is the most fundamental part of the systems software. It is always stored permanently in ROM storage and is operational as soon as the system is turned on. It permits the user to load additional software from disk or tape, which prepares the system for running applications programs or otherwise interacting with the user.

The systems software, as we have defined it here, also supplies the system with a "higher-level language" that is used to perform many tasks, such as creating more software, executing applications software, and generally "talking to" the CPU without using the much more difficult machine language of the microprocessor chip. Some common higher-level languages are PASCAL, BASIC and FORTRAN. BASIC is the one most commonly used for instructional programs. Any application written in BASIC or any other higher-level language will be compatible with any microcomputer system that is supplied with the appropriate version of the higher-level language in which the program is written. This assumes that the program has no machine language components and does not have to refer to a particular systems software in order to "read" or "write" data from external storage.

One of the limitations of microcomputer systems for educational use that we discuss in more detail later is the lack of compatibility among systems. As we just saw, the higher-level language capability of the system is one major determinant of compatibility. Besides this feature, the most important feature of the systems software in determining compatibility with other systems is the program that controls the operation of the tape or disk hardware. All applications programs that require information to be "read" from or "written" to external storage will be dependent on this software and will not run on a system with different systems software.

TYPICAL PHYSICAL CONFIGURATIONS OF COMPONENTS

The CPU, power supply, and principal interfaces are usually supplied in the same unit, often with a fan for cooling. Some systems are built with the CPU, a screen, the keyboard and an external storage system also in this single unit, the printer being the only separate component.

The keyboard and the CPU (with power supply and interfaces) are often built as a single unit with screen, disk drives and printer separate.

MORE ABOUT COSTS

Depending partly on the amount of storage, the CPU, with a power supply and interface system, can cost less than one thousand dollars and up to $3,500.00 or more.

Many business and education systems without a printer cost from about $1,300.00 to around $8,500.00, depending primarily on the amount of CPU storage, and the type and amount of external storage. Hard disk systems can increase this cost considerably. Printers vary in cost from several hundred to several thousand dollars, depending mostly on print quality and speed.

These figures are intended to give some idea of the range of cost of microcomputer systems, but they are not indended for serious budgeting. A specific system that can be purchased for approximately $6,000.00 is described in Part C of Section V.

If a computer consultant or programmer is needed to create customized applications software, the cost of a system can easily more than double.

ADDITIONAL COMPONENTS

Besides the components of a typical microcomputer system described above, many additional input and output devices are available at a reasonable cost. These "peripherals" are substantially cheaper for the popular education systems than for systems primarily used by businesses. Some examples of relatively inexpensive peripherals available for systems like Apple are:

- graphics boards that input drawings or tracings so that they can be manipulated, and calculations can be done on their shapes and sizes.
- light pens for interacting directly with information on the screen (as in responding to multiple choice questions).
- systems for creating and playing music.
- devices that read mark-sense test forms, bar codes, punched cards. One recent British product even enables the system to read short sequences hand printed on a small pad.
- interfaces that allow computer programs to interact with programs recorded on video disks.
- interfaces that operate electro-mechanical devices.

- an inexpensive phonemic voice synthesizer for Apple computers.

In exploring the potential of microcomputer systems for educational use, we say very little about these "bells and whistles." We look at a few of them again in Part C of Section V in regard to possible uses that the SWRL composition project might have for them.

APPLICATION PROGRAMS

By "applications programs" we mean all software that does useful things outside the system itself. For instance, application programs enable microcomputer systems to present instruction, perform calculations, edit text, keep and manipulate records, fill out forms, and simulate complex real world systems. Applications software is distinguished here from the systems software described earlier, which performs functions within the system. Most applications software is acquired on tape or disk, copied from printed listings, or created by the user.

Applications software that is intended for instructional use is sometimes termed "courseware", but we do not use that term here because, in our view, much of the software intended for business and other uses also has potential instructional uses and the term "courseware" is narrowly associated by some with a particular type of ("programmed") instruction.

SECTION III. LIMITATIONS

In this section we examine some aspects of current microcomputer hardware and systems software that appear to place limits on the use of microcomputers for instruction. Then in Section IV we look at the potential of microcomputer systems for instruction. In the current state of the art, there are two principal kinds of limitation on the use of microcomputers for instruction: their complexity of operation and their lack of intercompatibility.

COMPLEXITY OF OPERATION

It is tempting to conclude from the advanced state of the art of computer hardware and its recently much reduced cost that all that stands in the way of more general classroom use of computers is the lack of appropriate applications programs. However, there are serious limitations on the usefulness of computers as an instructional tool. One of these is that relatively complex new skills must be learned in order to operate microcomputer systems. Some teachers have a general impression that computer technology is inaccessible, and any complexity of operation appears to reinforce this impression. In any case, it is a common observation in discussions of the future of computers in the classroom that many teachers experience considerable anxiety when confronted with this tool.

New equipment is more likely to be well received when it is acquired in response to current needs and when the complexity of the routines for operating the system is minor relative to the importance of the instructional function it fulfills. In these respects, the introduction of microcomputers into a classroom often presents substantial problems. Microcomputers are inherently complex machines, but at the same time their potential for meeting instructional needs is very great. A major challenge to the developer of microcomputer-based instruction is to find ways of simplifying the operation of the equipment while at the same time exploiting its potential.

While addressing this challenge, the developer need not compound the difficulty by designing instruction which is more innovative than required by the new computer medium. The new computerized instruction can be designed to fit into current instructional practices without requiring these practices to be substantially modified. Furthermore, the new computerized instruction can be limited to satisfying needs that are already recognized, either implicitly or explicitly, by students and teachers. If the large potential of the computer for solving global instructional problems eventually leads the instructional developer to feel that teachers and students can be better served by substantial innovations in the overall instructional program, then the developer might consider two stages of computer-based instruction. The first stage, to be introduced at the same time the new technology is introduced, would require only minimal changes in non-computerized instruction and would clearly fulfill currently recognized instructional needs. Then the second stage, postponed until teachers and students had become familiar with the new tool, could introduce computer-based components that presuppose changes in the overall instructional system.

Whatever else designer/developers do to meet the challenge of introducing computers into the classroom, they must accomplish three basic goals. First, newly introduced computer-based instruction must meet currently recognized needs. Second, it must be integrated into the overall instructional system. And finally, it must be useable with minimum attention to hardware and software operating requirements.

Now let us turn to a more immediate kind of complexity introduced by the microcomputer and suggest something that might be done about it. Let us assume that the educational user wants to use a microcomputer system for an application such as word processing, keeping and sorting records, or interacting with a computerized component of an instructional program. In order to perform one of these operations on a typical microcomputer system with disk storage, a student or teacher would have to:

1. Turn on the system
2. Place an appropriate disk into the disk drive
3. Type in a command to load software into RAM
4. Press a key to execute the command

Besides these four steps, some of the following steps may be required:

5. The command in step 3 may load only systems software and another command may be required to load the desired application.

6. In some cases, the command in step 4 loads a programming language that is necessary for running the application, and still another command is required to load the application itself. This last command may require two steps, one to put the system into a command mode and one to load the application.
7. Once the application is loaded, another command is typically required to execute it.
8. An additional command is required when changing to a different application. If the software required for the new application is not on the disk (or disks) currently in the disk drive(s), then the appropriate disk must be found and placed in a drive before the new command to load is issued.

One of the apparent problems for the teacher or student is remembering the various commands. Another is the intrusiveness of these computer operations in the normal flow of instruction.

Systems that are popular for classroom use, like the Apple, TRS-80 and PET, eliminate the need for steps 2 through 4. When these systems are turned on, a BASIC language capability is immediately available. This BASIC is customized for each of the systems to allow applications such as drawing graphics on the screen to be controlled directly from the BASIC software. Further applications are loaded from disk in the usual way, and in some cases the load command also causes the application to execute.

There is no reason why a regular business microcomputer cannot be made just as simple to operate as the Apple, TRS-80 and PET. In fact, such a system could easily be customized for some particular set of educational applications in such a way that it would be even easier to use for those purposes than an unmodified popular classroom system. One way to accomplish this would involve the following two fairly simple steps.

a. Program a PROM (a programmable ROM) so that, when directly plugged into or otherwise added to the system, it causes a larger special program to be loaded from disk when the system is turned on. The hardware to program the PROM is not expensive, but in any case there are services that do this. Some microcomputer systems are supplied with a PROM programming feature.

b. Program the larger, disk-based special program so that when it is automatically loaded it will find, load, and start a particular instructional application. One feature of this special program could be to display a menu of available applications with an invitation to type in a choice. The special program would then execute that choice. Any helpful prompts that might facilitate this process could be included. For instance, if the chosen application were not on the disk(s) currently in the disk drive(s), a prompt could give the name or number of the required disk. This step (step b) could be accomplished simply by very straightforward programming.

THE COMPATABILITY PROBLEM

The principal limitation on widespread dissemination of high quality microcomputer programs is probably the lack of compatibility among systems. The principal functions of microcomputer systems that lack standardization or otherwise create compatibility problems are:

1. Machine Languages

Each microprocessor chip has its own way of dealing with the electrical impulses by means of which the CPU does its processing. Three chips currently dominate the microcomputer markets of interest to us here, the Intel 8080, the Zilog Z-80, and the MOS Technology 6502. The machine language instructions for the Z-80 are a superset of those for the 8080 and so the latter is "upward compatible" with the former, but the 6502, which is used in Apple and Pet systems, has a distinct machine language.

All microcomputer processing is done in machine language. Higher-level language instructions must be translated into the machine language of the particular system before they can be executed. For this reason, programs written in machine language are more efficient than higher-level language programs. Some types of application such as word processing and complex sorting operations are not practical unless written in a particular machine language, and so these programs are system dependent.

2. Disk Operating Systems

An important part of the systems software of each microcomputer system is the DOS, or disk operating system. It enables the system to read and write information to and from external disk storage. Unfortunately, there is no standard for disk operating systems. Both the disk hardware and the associated systems software create compatibility problems.

3. Interfaces

Some systems do not have the necessary interfaces to make use of various types of useful peripheral equipment, although such interfaces can frequently be added to the system without excessive cost.

4. Higher-Level Languages

BASIC, PASCAL, and other higher-level languages are independent of the machine languages they implement. It is only necessary for the particular system to have a resident program to translate them (technically either "compile" or "interpret" them). But there are various "dialects" of each language that cause compatibility problems. Also, some manufacturers add idiosyncratic features to the higher-level language supplied with their system, causing programs containing those features to be exclusive to the particular system. A more fundamental incompatibility results when a program is written in a version of a higher-level language that requires too large an interpreter for the RAM size of a particular system.

The compatibility problem appears to have caused some purchasers to select certain popular components over others in such a way as to informally establish a sort of standard system. This has happened in the business market and to some extent among serious hobbyists, but not in the education market, which occupies the lower end of the price range and is not yet sensitive to any widespread availability of software. Thus we find three highly idiosyncratic systems- -Apple, Pet, and TRS-80- -in the classroom.

The typically somewhat more expensive "standard" microcomputer is worth our consideration, however, both because of the applications, like word processing, that are available to it, and because a modification has recently become available for Apple which adds to it some of the important characteristics of the "standard" system. In Section V, we see how the SWRL project can have both Apple and "standard" compatibility.

It should be noted that the "standard" system to which we refer here is by no means an official standard, nor does it have many completely compatible representatives. It is a set of components which have gained market acceptance and for which many valuable applications have been written. Systems may have some or all of these components. Many otherwise excellent systems are not not compatible with this so-called standard system.

The principal characteristics of the "standard" system are:

1. It uses C/PM brand or C/PM-compatible systems software because this is by far the most widespread operating system.
2. It uses an 8080 or Z-80 microprocessor because C/PM requires one of these.
3. It typically has two floppy disk drives. There is an IBM standard format for writing data onto these disks, but no standard of data density, so this remains an obstacle to wider compatability. Material can be copied from one disk to another via an interface between disk drives or from one machine to another using the ASCII standard data-transfer format used for telephone data transfer.
4. It has an ASCII standard typewriter-like keyboard. The Apple keyboard, for example, is not standard.
5. The screen displays 80 columns by at least 24 lines of characters. The Apple, for instance, displays only 40 characters across the screen.
6. It has an S-100 bus interface system to take advantage of the large amount of peripheral equipment (e.g. printers) that are compatible with this interface system. This characteristic is not essential for compatibility, but desirable. The Apple, for instance, must be modified to accept some S-100 compatible peripheral hardware.

We return to the compatibility problem in Section V.

SECTION IV. THE POTENTIAL OF MICROCOMPUTERS FOR INSTRUCTION

In this Section, we consider both the general instructional potential of microcomputer systems and specific types of instruction that it might be useful to computerize. This discussion is directed both to the general need for computerized instruction and to the specific case of the SWRL composition project. SWRL's selection of specific aspects of composition instruction to computerize will follow from an analysis, which is currently in progress, of the types of instruction needed for composition skills. Meanwhile we attempt to anticipate the interests of SWRL project. The ideas included here should be considered only as beginning points for stimulating discussion among instructional researchers and developers.

This Section in in three Parts: Part A, Microcomputers Versus Larger Systems; Part B, Existing Computer-Based Instruction; and Part C, Unrealized Potential.

PART A. MICROCOMPUTERS VERSUS LARGER SYSTEMS

With the advent of hard disks and networks that share disk storage, microcomputer systems can perform many tasks that were until very recently limited to larger systems. Advantages of microcomputers over the larger systems, besides cost, include the following:

- The microcomputer is very flexible. It can function on the one hand as an independent computer, or on the other hand as a member of a distributed network, sharing storage with other microcomputers or acting as a timesharing terminal in a system that includes a large computer.

- When a user employs several microcomputer systems instead of a single larger system with terminals, the components of the microcomputer systems are interchangeable, and it is unlikely that all systems will be inoperative at once. All systems must be expected to be inoperative from time to time.
- A microcomputer system that uses "memory mapped video" to deliver information to the video screen (using a procedure called "direct memory access") is faster for some word processing operations than one that delivers this information to a terminal. Memory mapped video also avoids a screen flicker that some users find objectionable when doing microcomputer word processing. This type of interconnection between the screen and the CPU is found in many microcomputer systems but it is not used on larger computers. A disadvantage of this feature is that such a system does not readily accept substitute terminals.
- Wher "timesharing" on a larger computer (the standard mode of interactive operation on a large computer) the user sometimes experiences substantial slowdowns in operation and even has to wait to get on line. This problem was the main complaint of teachers using the State of Minnesota timesharing system and contributed to a decision to

include microcomputer services in the Minnesota Educational Commputing Consortium.

- Many powerful applications that require peripheral hardware are available for use with microcomputers at a small fraction of the cost of similar equipment for larger computers. For example, graphics, voice synthesis, video interface, and music composition are not usually thought to be practical for instruction based in larger computers, but are available at more reasonable costs for use with microcomputers.

This is not to say that larger computers can be replaced. For very large record-keeping and accounting tasks, for tasks that require extensive calculations, for extensive timesharing, and for many other purposes, microcomputers cannot compete with full scale computers.

PART B. EXISTING COMPUTER-BASED INSTRUCTION.

It is generally agreed that the current state of the art of computer-based instruction is primitive. Much of what is available is an adaptation of the stimulus-response format of the programmed instruction of the 1950's and 60's. Computer games have also been adapted to instructional purposes. Computers are used successfully in both elementary and secondary schools to teach about themselves--a subject referred to as "computer literacy". Some computerized math and science instruction is quite successful and a few very creative applications have been developed for other subjects, particularly at the college level.

A few programs are offered through established commercial sources. For instance, California Computer Corporation in Palo Alto offers drill and practice programs with extensive record-keeping features in a variety of elementary school subjects. Their system is based in a larger (mini)computer with timesharing terminals. This system has been used successfuly, for example, by Los Nietos Elementary District, in Santa Fe Springs, California, for a number of years. Microcomputer-based programs, on the other hand, are typically produced by individuals, usually by teachers for their own use, and disseminated, if at all, through various informal networks such as clubs, newsletters, features in computer magazines, etc. These microcomputer-based instructional programs typically take the form of drill and practice and instructional games.

According to one view, computerized instruction cannot be expected to become systematically integrated into overall instructional programs nor can it be expected to be of high quality, unless it is created by a large-scale instructional development effort, and it is not currently profitable to marshall such an effort. In any case, widespread high quality computer-assisted instruction does not exist. The reason should be apparent from the limitations of microcomputer systems discussed earlier in this report, especially the compatibility limitation.

The present report is not a review of the state of the art of computer-assisted instruction. We are concerned here with the instructional potential of microcomputer systems. Once SWRL has a microcomputer system, we will be able to review current programs, but from what we have seen so far, most development work lies in the future. It remains for future computerized instruction efforts to seriously address instructional needs beyond the need for students to learn about computers themselves or to be motivated by the novelty of computer equipment to participate in programmed drill and practice.

PART C. UNREALIZED POTENTIAL

As contrasted with current practices in computerized instruction, potential applications seem almost limitless. Even if we were to consider only the classic type of computer-assisted instruction, adapted from the once popular programmed instruction used in teaching machines and programmed textbooks, it is now possible to program much more sophisticated sequences, sensitive to more variables in the instructional environment. For example, at any step in the instruction the computer could react differently to the same response from two different students, depending on differences in the students' academic profiles, cultural backgrounds, bilingual proficiencies, etc. This highly personalized instruction could then be supported by automated audio, video, color graphics, animation, electro-mechanical devices, or other "bells and whistles".

In this Part, we discuss some specific ways of using the computer to deliver instruction. These suggestions are not intended to be definitive. They are intended as a means of starting a dialog within composition instruction staff and elsewhere about the potential of microcomputer systems for meeting instructional objectives.

1. Programmed Instruction

One of the most common uses of computers for instruction derives from the once popular "programmed instruction". In the beginning, computer-assisted instruction (CAI) assumed the stimulus-response-reinforcement format of programmed instruction. This method had been used in the fifties and sixties with programmed textbooks and teaching machines to deliver instruction in steps or "frames," a technique consistent with popular learning theories of the time. Computers lend themselves easily to this instructional format: They can deliver a stimulus on a screen or in other ways, accept the student's response from a typewriter keyboard or other input device, and compare the response to a stored set of possible responses. Depending on the response, the computer can then supply appropriate reinforcement. Also, the computer program can choose the next "frame" of the instruction to suit the student's response.

In spite of the fact that programmed instruction is no longer as well received as it once was, a large part of current computerized instruction uses this format. A typical use is for multiple-choice questions with loops back through previous instruction when incorrect responses are given. The programmed-instruction format is encouraged by programming aids such as the higher-level language, PILOT, that make it easy for teachers to author CAI. Existing CAI programs in the programmed instruction format do not typically exploit very much of the potential of the microcomputer. The program often requires only a "yes" or "no" response and seldom allows a choice among more than three or four alternatives.

One kind of unused potential in this format is evident in everyday use of the SOL microcomputer at SWRL. This modest 32K RAM microcomputer system can find a requested sequence of up to forty characters within a text of over two thousand words and display it on the screen without any perceptible delay. An adaptation of this feature (which uses machine language) would permit the SOL to almost instantly react to student responses up to forty characters long after comparing the response with at least three hundred possible responses. This is probably enough comparison to permit the instructional designer to allow complete freedom of response to carefully selected stimuli, producing a very natural dialog between the student and the computer. In discussions of other types of instruction below, this technique of comparing responses with a file is pursued further.

Authors of programmed instruction and other CAI have begun to embellish it with the various "bells and whistles" that have recently become available for classroom microcomputer systems. These permit the reinforcement step of classical CAI to be in the form of musical tones, synthesized speech, graphics (including color), and animation. These "bells and whistles" are particularly suited to the next type of instruction we examine: instructional games.

2. Instructional Games and Simulations

Instructional games are a recent spinoff from the home computer games market. Most of these computer games can be thought of as simple instances of the very sophisticated technique of computer simulation. In general, anything for which a precise model can be constructed can be simulated by computer. Games are typically used in instruction as entertaining ways of practicing basic skills. More sophisticated computer simulations display complex processes such as ecological, physical, or social systems and allow the user to interact with them.

A composition instruction program could probably benefit by constructing computer simulations of language processes, although this effort may be beyond the scope of the SWRL project. For instance, one could determine a few of the more important variables involved in communicating a description of an object or process. A computer simulation of this process could be modelled after research in "convergent communication" where the process is typically oral and the participants have a screen between them, so the communicator is not able to refer visually to the object or process. In our case, the computer would be one of the participants and student responses would be in writing. The need to communicate without diectic reference is even more important to writing than to oral communication. The suggested convergent communication could designate a writing task related to an actual object or process in the student's possession. To keep the simulation sufficiently simple, the task could be directed by prompts that would guide the student's responses to keep them within the bounds of a model. The computer could then evaluate the effectiveness of the composition according to criteria in the model, and provide instruction in those aspects of convergent communication the student did not manage well.

3. Personalized Instruction

The computer typically personalizes instruction by reacting appropriately to the student's particular responses. Another dimension of personalization could be added by supplying the computer with information about the student other than his or her responses to the instruction. The teacher might want to keep student records in a computer file in any case. At the elementary level, for instance, the typical student record could be extended to include such personal attributes as favorite color, brothers and sisters, pets, hobbies, etc. The computer could then engage in a personalized dialog with the student that would require the student to make relatively complex written responses. It seems likely that the specialist in composition instruction could make use of such dialogs for instruction.

Keeping and using files of the type envisioned here is a normal data management function of computers. A particular student record would be of a size that could easily fit into microcomputer RAM along with the instructional program and working space for interactive processing.

More complex versions of such a program might require reference to small dictionaries of words and phrases stored on disk that would be moved in and out of RAM as needed for comparison with student responses. Systems with rapid access to large amounts of external storage (hard disk storage) could handle comparisons to lists of thousands of alternative responses. As indicated in the discussion of programmed instruction, the semantic domains of these comparison files would have to be limited to stimulus domainsth relatively well bounded sets of possible responses. For personalized instruction, relatively small domains such as colors or breeds of dog would be useful. A computer statement like, "Sorry, I don't know what '--' means," could take care of unusual responses.

A personalized dialog program such as this could make use of existing word processing and list-processing software. Word processing would enable the student to construct and edit complex responses, while list-processing would insert personal references into the computer's side of the dialog. Special routines could enable the computer to react appropriately to kinds of pets, numbers of brothers and sisters, and other information in the student record. For a composition program, learning could be centered in the process of creating responses.

One value of this type of personalized instruction would be that instructional programs could be tailored to particular cultural backgrounds. The references, motivating forces, and familiar contexts of a particular culture could be made available selectively, depending on the background of the student using the program and his or her personal reactions to the included cultural features.

A principal limitation on this and many other possible instructional uses of the computer is the size of the effort required to design and program the application. One possibility would be to employ existing software for major parts of the effort. For instance, a readily available list-processing program is sufficient for personalizing computer messages by a process similar to filling in the address and salutation on form letter. But, in general, most of the suggestions made here would require a large amount of original design and programming.

4. Creative Writing with a Word Processor

One of the principal reasons for including an investigation of microcomputer applications in the current SWRL composition project was the possibility of making available to students the creative potential of word processing. There is no question that word processing is a powerful tool that could, under the proper circumstances, be a powerful aid to instruction. However, it is not as obvious as it might seem how this tool should be included in composition instruction. In order to appreciate this problem, we need to take a closer look at the possible functions of this tool in the general society.

Word processors are rapidly becoming a necessity in the business world. Although there is some resistance to the changes they impose on secretarial staff, they will ultimately replace other methods of typing, editing, and formatting documents. Educators can be certain of a coming need for training in word processing skills, at least for secretaries.

Secretarial use of word processors is not the same as using them to create original documents, however. Writers (authors, professionals, managers, etc.) have different word-processing requirements than secretaries have. Secretaries are typically concerned with converting an existing document into a finished product. Here, straight typing and powerful print-formatting functions are most important. The editing needs of secretaries are more mechanical than those of writers, so the secretary can tolerate more complexity in exchange for better formatting capability, even though this may intrude into creative editing. Writers need a somewhat different set of functions in a word processor than those needed by secretaries. Writers need to have a broad overview of a document in progress. For this they need as much text on the screen at one time as possible, or even on multiple screens. Other important features that help maintain an overview of the writing task include space for the entire document in RAM, the ability to "scroll" very quickly from one point to another in the document, and a "search" function that will instantly find any specified heading or other phrase in the document. Writers need powerful, simple-to-use editing functions such as insert, delete, and block movement, but complex print-formatting operations only get in the writer's way.

Turning to the use of word processors in composition instruction, one of the first questions is, would it be desirable to require students to learn to type as a prerequisite to creative writing? One current argument says that computers with typewriter keyboards will be so ubiquitous within a few years that there is now a basic need for "computer literacy," which includes typing skill. Whether this is true or not, one straightforward use of word processors in instruction would be to familiarize students with computer operation and to teach secretarial and typing skills. They could also be used as a powerful tool to facilitate complex writing tasks, but students would first have to learn typing and become familiar with a relatively complex set of computer operations.

Once the decision is made to use word processors in instruction to facilitate the task of the writer (as contrasted with the secretary), then authors of computerized instruction have a powerful tool for guiding the composition process. They might want to consider programming the computer to be sensitive to the syntactic and semantic content of the composition. This is possible within limits. It is not possible for the computer to judge the appropriateness or effectiveness of unconstrained creative input, because the grammatical and other determinants of these characteristics are complex and poorly understood. It would probably be possible, however, to formulate writing tasks that are sufficiently constrained to permit computer evaluation of the syntax and even some of the semantic content of relatively complex responses.

One possibility for the SWRL project would be to program existing word processor software with prompts to guide the student through steps in a compositon task. This would bypass the large programming task involved in writing word processor software and permit the project to focus on questions about the effectiveness of different ways of using this tool in writing instruction. It might even be posssible to make some of the prompts sensitive to the student's syntactic or semantic input to increase the instructional power of this approach. There is available a particular word processing program called Magic-Wand, which lends itself to this type of customization.

In deciding to use an existing word processor program--software that is typically written in machine language and delivered on a disk--one must consider the compatibility of the software with the system on which it is to be used. Some word processor programs will run on a variety of systems (e.g., WordStar and Magic-Wand), while others are completely machine dependent (e.g. the Vector Graphic system and probably the forthcoming Apple III system). Some of those written specifically for Apple II do not operate like a standard system because they have to work with a keyboard without lower case letters and a screen width of only 40 characters. Programs written for Apple II include Supertext, Write-on II, Apple Writer, and Easy Writer.

At present, the few good word processor programs that are available for use on more than one microcomputer system operate under CP/M systems software and are written in machine language for 8080 and Z-80 microprocessors. These microcomputer features will be recognized as characteristics of the systems described in Section III as the largest group of relatively intercompatible systems currently available.

SECTION V. ACQUIRING A SYSTEM

The main purpose of this Section is to help SWRL staff decide what equipment to acquire for the composition project. However, in order to put the specialized needs of the current SWRL project into perspective and to make this discussion useful to possible future SWRL projects, we first consider the equipment needs of computerized instruction projects in general. These general requirements are discussed in Part A. In Part B we focus on critical issues that must be addressed in meeting these needs.

PART A. GENERAL REQUIREMENT

1. The designer/developer will need to evaluate current practices in computerized instruction.

Most projects that design computerized instruction, including the SWRL composition project, will need to evaluate current instructional programs, authoring aids, and a variety of software and peripheral hardware that might implement design objectives. This cannot be accomplished with any single microcomputer system currently available.

This difficulty is one instance of the compatibility problem, which was discussed in Section III. The compatibility problem is discussed again in Part B of the present Section.

2. The designer/developer will need to produce programs that are compatible with whatever systems will be in use in the schools when the programs are implemented.

This requirement adds a dimension to the compatibility problem. The designer/developer's system must not only be compatible with current applications software, but also with hardware that will be in use in three to five years. The technology is changing so rapidly that it is difficult to design programs now that will be useable on equipment purchased even two or three years from now. This is particularly important to the private publisher, who must expect to distribute large numbers of programs to justify the expense of producing quality computerized instruction.

As we noted at the beginning of this report, many school systems, private companies, research facilities, and others involved with microcomputer-based instruction are currently using the Apple II, TRS-80, or PET systems. Our incomplete evidence indicates that the largest commitments are to the APPLE II system (e.g., the commitments of Minnesota and North Carolina, Region IV in Texas, and the Bell and Howell Corporation).

In attempting to anticipate future technology, designer/developers must either rely on current systems being upward compatible with more powerful systems already being developed or they must expect to translate parts of their programs at the time they are disseminated.

3. The designer/developer will need to try out newly designed programs.

Tryouts can be conducted in house or at a school using equipment provided by the designer/developer, or they can be carried out on school equipment, provided it is compatible with the program to be tried out. For some types of tryout it might be desirable to simplify the equipment with a keyboard overlay that allows only certain keys to be pressed, but with this possible modification, any equipment compatible with program software could be used. Tryout considerations should probably not influence the choice of equipment to be used by the designer/developer except to the extent that some cooperating district that has compatible equipment must be available within a reasonable travelling distance.

4. The designer/developer will need to consult at a distance with districts that are trying out or using a new program.

When consulting by mail or by phone about details of program implementation it is useful to have equipment in house that duplicates the operations being performed at the school.

5. School requirements are distinct from the requirements of the designer/developer.

Many classroom requirements are not relevant for the choice of equipment on which to author instruction. For example, for classroom use, any features that make the equipment available to several students at one time are of particular interest. In one instance (the Telemath computerized mathematics instruction program, which was developed in the San Diego Unified School District), microcomputer hardware was designed that would accept input from two keypads, permitting two students at a time to use each system. Another example of a hardware adaptation for the classroom that would not be needed for authoring instruction is the systems simplification suggested in Section III of the present paper.

PART B. ISSUES TO CONSIDER WHEN ACQUIRING A SYSTEM

Complexity

System complexity will not be a problem for the designer/developer unless it reflects commitment to a system that is not easy to simplify for classroom use. Within reasonable limits, complexity should be accepted by the designer/developer wherever it makes the system more useful for the design and development of instruction. For example, it would be necessary to accept additional complexity in order to take advantage of the recent hardware that permits addition of a Z-80 chip to an Apple II system. In return, this would make the system compatible with a wider variety of applications, and programs could be designed on it that include the machine language of either of two widely used microprocessors.

Compatibility

This is the major issue for the user who wants to take advantage of a variety of available software or for the designer/developer who evaluates such programs, authors programs and disseminates them widely, tries out programs, or consults with users that may have a wide variety of equipment. Compatibility has been a consideration in most of the discussions in the present document and is the principal consideration for selecting the particular system described in Part C of this Section.

Reliability

Both inherent reliability and the availability of maintenance are important considerations for both schools and designer/developers. Mechanical equipment (as opposed to purely electronic equipment) is the most subject to failure. This includes printers, keyboards, graphics tablets, and disk drives. Printers are probably the greatest source of trouble, especially letter-quality printers. A reliable dot-matrix printer can be used for most development work. It is worth noting that the keyboards of the popular classroom machines typically have a much shorter predicted life span than those of many "business" oriented microcomputers, although this may be more important for the classroom, where the keyboard is likely to be abused, than for the designer/developer.

Timing the Acquisition

As this is being written, at least two new hardware introductions of particular importance to classroom applications are imminent. Similar situations will probably continue to exist, making the timing of equipment acquisition very difficult. These two developments are probably more important than most because of their possible effect on the compatibility of the system one might acquire at this time.

1. Apple Computer Inc. is introducing a new, model, the Apple III. It was demonstrated at the NCC computer show in April, but is not

yet in dealer showrooms. Of particular interest to the instructional developers is the fact that Apple III can be set to emulate the popular Apple II. It is not clear yet, however, what will be involved in getting the Apple III to accomodate the plug-in modifications now available for the Apple II. Recent word is that this will not be possible. Besides its compatibility with Apple II, advantages of the Apple III include:

- standard keyboard and 80-column screen forword processing;
- Much larger RAM.

2. A "Z-80 Softcard" has been designed for Apple II to allow it to run 8080/Z-80 machine language programs and the popular CP/M systems software. This should be arriving at dealers very soon. In Section III, we described this machine language and the CP/M systems software as characteristics of the most generally intercompatible group of microcomputer systems on the market.

APPENDIX F

A PROPOSAL FOR AN ELEMENTARY SCHOOL COMPOSITION PROGRAM

Ann Humes

A PROPOSAL FOR AN ELEMENTARY SCHOOL COMPOSITION PROGRAM

BACKGROUND INFORMATION AND RATIONALE

Until recently, educators and researchers have been more interested in students' ability to read than in their ability to write (Graves, 1978; Gundlach, 1980). But this disparate focus is changing, stimulated by the results of community, state, and national assessment projects (e.g., National Assessment of Educational Progress, 1980) that have substantiated what teachers, parents, employers, and students have regarded as self evident: Most people are not proficient writers.

An important reason for "the writing problem" is that instruction has traditionally emphasized the product of writing--what students write--rather than the process of writing--how students write. Despite recent scholarly support for instruction in writing as a process (e.g., Shaughnessy, 1977; Graves, 1978), few instructional programs attempt to teach the composing process. Furthermore, the few programs that do try do not have a unified approach: Although they teach some composing processes and tasks, they present component skills separately. This approach is not satisfactory. It will not ameliorate "the writing problem."

This proposal for an elementary school composition program fills this need for a unified approach to the teaching of writing in the elementary school (grades 1-6). The proposed program teaches the composing process while teaching the component skills required to complete a writing task within the context and content of the process. Students attend to considerations of the writer's purpose and audience while learning to plan, draft, revise, and edit their writing. When a skill is necessary for a writing task, it is taught in the context of that task. Thus instruction on component skills (e.g., punctuation and capitalization) does not interrupt the writing process since it is integrated into that process.

Consequently, the program outlined here is a complete written-language program. With the exception of spelling and handwriting skills, all the skills students need in order to write competently are covered--punctuation, capitalization, language usage, sentence structure. Spelling skills are not included because students need so many spelling skills that separate spelling instruction is appropriate and necessary. Manuscript and cursive writing skills are also critical to composition, but they, too, can be taught in a separate program. Instruction on grammatical terms--on naming, for example, subjects and verbs--is not treated because research has shown that teaching grammatical knowledge does not improve students' writing (Braddock, Lloyd-Jones, and Schoer, 1963).

CONTENT OF AN ELEMENTARY SCHOOL COMPOSITION PROGRAM

The proposed content of a composition program for grades 1-6 includes products (see Attachment A), activities and techniques (see Attachment B), and component skills (see Attachments C and D).

Products

The result of a composing process is an actual discourse product--an essay, a story, a poem. The products in this proposal are categorized under discourse modes and format types.

Discourse modes comprise one classification because the focus of the proposed composition program is on the proces rather than the product of writing: Writing is a cognitive-processing activity, and discourse modes focus on the specific ways of thinking about a subject; each of the modes has its own logic, as well as its "own organization patterns, and, to some extent, its own stylistic characteristics" (Kinneavy, 1971, p. 37). Correspondingly, classification by the discourse mode dominant in the writing task is appropriate because the requisite skills are relevant to writing tasks beyond a single, specific product type. Other, albeit controversial and complex, classifications for product types might be more theoretically satisfying (e.g., Moffett, 1968; Kinneavy, 1971; Britton et al., 1975). However, these classifications entail specification of parameters for audience, purpose, and writer's persona. This specificity would occur in the actual instruction rather than in the somewhat abbreviated architecture described here. Additionally, the discourse modes are relevant across product types, so the knowledge and skills acquired are more easily generalized to other writing. Furthermore, the classifications used here are readily comprehensible to practitioners and students because of their practical, non-theoretical nature.

The discourse modes proposed for an elementary school composition program are Storytelling, Describing, Informing/Explaining, and Persuading (see Attachment A). Storytelling covers both personal narratives and fictional (climactic) narratives, including plays. Describing includes composing word pictures of objects, animals, people, and scenes, with the describing tasks being subsumed at the upper grades under the stories students write. Informing/Explaining tasks are writing directions, summaries, analyses, and reports, as well as writing in school subjects such as math and science. Persuading covers reward-promising announcements, letters to the editor, and arguments.

Format types comprise the other classification (see Attachment A). Products categorized here are distinguished more by their format than by their discourse mode. These format types include letters (both personal and business), practical forms such as applications, and poetry.

Activities

<u>Preparation</u>. Preparation includes activities that teach students the component skills needed to compose a product. While studying examples of the kind of products they are to write, students attend also to the component skills introduced by the product. For example, when students are to write a story, they practice the syntactic structures that are frequent in narrative writing; when they are to write directions, they practice writing and punctuating imperative sentences within the context of a set of directions.

<u>The Composing Process</u>. The major activities of the composing process include Planning, Drafting, Revising, and Editing (see Attachment B). These subprocesses are listed separately for convenience in discussing and, sometimes, in teaching them. However, this separation does not imply a linear process: Planning is not complete before some Drafting begins; all Drafting is not finished before writers revise; all Revising is not completed before any Editing begins. Writers move back and forth among these subprocesses. For example, after text has been drafted, a gap may be noticed and new content additions must be planned.

1. Planning.

Planning includes conceptualizing the elements of the task. Writing goals are set and then ideas are generated. This idea content may be generated from external sources or from the writer's mind. When writers search their own minds, they often use heuristic probes for generating content. In the Composition Program, these probes are generally comprised of sets of questions students ask themselves in order to generate content. For example, when students first learn to generate ideas for describing something, they learn to probe their own minds with this heuristic:

What does it look like?
What does it sound like?
What does it feel like?
What does it smell like?
What does it taste like?

Such heuristics not only generate content for use in composing, but they also teach students how to probe their own minds for ideas.

During Planning, students also study the kind of content that is characteristic of the product they are to write. For instance, students learn that the content for invitations is what, where, and when information and a request for attendance; the content for directions must include procedural steps. Instruction on planning also includes instruction on the process of arranging content in order to give form to a final product, as well as on the arrangement plans that are appropriate for the concomitant writing task.

2. Drafting.

Drafting is putting ideas into visible form--translating content from one form of symbolization (thought) into another form of symbolization (graphic representation). It involves putting words on paper and is synonymous with what is frequently called "writing." However, true writing also includes ~~the~~ mental activities that occur even when a writer is not putting words on a page. Students learn to draft without excessive concern for surface errors because such concern inhibits the writing process. They also learn important skills that enable them to draft with fluency and coherence.

3. Revising.

Revising is reformulating, making substantial changes to improve a text. Students learn to add new information, to change the order of content, to remove unrelated sentences and redundancy, and to vary the structure and length of their sentences. Students often resist changing words that are already on paper. The proposed composition program teaches students to overcome this resistance--it teaches them how to revise and what to revise, as well as to appreciate the beneficial results that revising has on a text.

4. Editing.

Editing in this composition program involves those activities that require students to change words or to correct spelling, punctuation, and capitalization. Students learn to proofread for errors and to edit them out of the text before writing the final version. Students are more likely to undertake serious reformation of their text when they learn that they may attend to surface errors after their minds are no longer preoccupied with making more demanding changes in content and arrangement.

Evaluation. Evaluation is analyzing a composition for its strengths and weaknesses. In the proposed composition program, it includes both peer critiquing and teacher comment. Peer critiquing is an important evaluative procedure because it enables students to observe how their their writing affects others (Beaven, 1977). It also provides insights about writing techniques (Moffett and Wagner, 1976). Students frequently will accept the evaluations of their peers more readily than evaluations by their teachers. The procedures for initiating and directing peer critiquing are described in the teacher's materials.

Teachers evaluate during the composing process by making comments that help students to write and rewrite better compositions. These comments focus on teaching, not testing. Teachers assist students by asking questions about the writing and offering suggestions for improving it.

When students have completed their final drafts, compositions are formally evaluated. The evaluation procedure proposed here is modeled after the SWRL method for evaluating writing samples (Humes, 1980). Teachers who have used this method, without formal training, have found that it makes scoring compositions easy, that scoring is reliable, and that scoring provides diagnostic information about the component content and form skills.

Publication. Publication provides the students' completed writing with an audience considerably broader than one teacher. Publishing is an important part of composition instruction because it gives students a sense of accomplishment; their writing is not simply graded and discarded. Students' compositions can be published by posting them on walls and bulletin boards, by incorporating them into books for a class library, or even by sending them to publishers of newspapers and magazines. When these publication methods are not feasible or are not appropriate, writing can be published by having students read their compositions to their classmates.

Techniques Fostering the Writing Process

Certain techniques facilitate developing the ability to compose. Each technique fosters some or even all of the subprocesses of the composing process and thus is critical to instruction on the composing process. These techniques include journal writing, word associating, clustering, freewriting, sentence combining, and matrix constructing.

Journal Writing. Students write at least three sentences each day in their own journals. They express themselves in their journals by writing about something they see, think about, are confused about, or want to complain about. Teachers do not look at the journals for correctness. In fact, teachers need not read the journals at all if they so choose. If teachers respond to the journals, they respond to the content (to what students write about), not to form (to the correctness of the writing).

Keeping a journal gives students important writing practice and gives students first-hand experience with writing as a way of communicating. Studies have shown that students who keep journals improve their writing over the duration of a school year (Staton, in Humes et al., 1981).

Word Associating. Word associating involves generating and recording words that are elicited by the presence of an object or by the name of an object, idea, or event. Associated words are written down during a one-to-five-minute period. During that time, students generate all the possible words they can associate with the target object, idea, or event. The range of word associations may be limited to a specific type of associating prescribed by the teacher or by students themselves. For example, associations may be restricted to the uses of a particular object; this is a content restriction. Another kind of limitation may be by word class--the student may elicit only words that describe the object in relation to the sensory experiences the object evokes (e.g., sensory words that evoke visual and tactile images).

Clustering. Clustering is a follow-up activity for word associating. Associated words are arranged into clusters of related content. This arranging may entail circling groups of related content. It may also involve drawing arrows from one content expression to another or by numbering the ideas in presentation order. When content is recorded on separate cards or papers, the clustering involves grouping the pieces of paper. After students group the related ideas into clusters of related content, they then determine a sequence for the various clusters of content.

Freewriting. Freewriting is a technique used to discover and explore ideas by writing (Elbow, 1973). It gives students regular practice in writing while not burdening the teacher with additional paper grading. The writer begins writing, usually motivated by a specific stimulus, and lets his/her discourse proceed in any direction so long as composing does not stop. If a block occurs, the writer repeats the last word until the block is overcome, or he/she writes, "I can't think of anything to say." The writing proceeds without concern for the amenities of written discourse (e.g., mechanics, spelling, complete sentences). Correspondingly, freewriting is not evaluated by the teacher.

Matrix Constructing. A matrix is a physical paradigm that can be used to generate and subsequently arrange content by focusing on facts that can fit into the cells occurring at the intersections of the horizontal and vertical categories of a topic. It is especially amenable to generating and arranging comparison/contrast content. Matrices can be used for generating content from either external sources or the writer's mind.

Sentence Combining. Sentence combining is an instructional technique that can enhance students' syntactic fluency. Students are given two or more simple sentences and are taught how to combine these sentences into one sentence. Yet no grammatical terminology is employed. Students become practiced in producing sophisticated syntax--syntax that they have already mastered in their oral-language production. Furthermore, the sentence-combining instruction proposed here is sequenced so that students practice the syntactic structures that appear with high frequency in the discourse mode they will use to complete a specific composing task. For example, when third-grade students are preparing to describe something, they can be given sentence-combining practice for compound sentences, since this syntactic structure appears frequently in descriptive discourse.

Skills

Many component skills must be employed in order to put words on a page. Consequently, drafting, while simultaneously functioning other elements of the process, makes huge demands on the writer's cognitive processes (Scardamalia, referenced in Bereiter, 1979). This mental load becomes less difficult as an increasing number of drafting skills become automatic rather than consciously driven. This move to "automation" is expedited by the instruction and practice in requisite skills of language, coherence, and form.

In the proposed composition program, these skills are presented when they are needed for the assigned writing task. As noted above, when third-grade students prepare to describe something, they receive sentence-combining instruction and practice on compound sentences because this syntactic structure is frequent in descriptive writing. To compose a compound sentence correctly, students must use a comma to separate the elements of the compound sentence. Thus this skill is presented simultaneously with the sentence-combining practice for compound sentences. Other examples further indicate the unified structure of skills and process in this proposal: When students learn to write map directions, they also learn to write and punctuate the imperative sentences that are characteristic of directions (e.g., <u>Turn left at the stop sign.</u>); when students learn to write personal letters, they learn the letter-format skills necessary to write personal letters as well as the prerequisite capitalization and punctuation skills (e.g., capitalizing the names of months, inserting a comma between city and state); when students compose a script for a play, they (1) use a comma after a noun of direct address, (2) use an apostrophe in contractions (since contractions are characteristic of the oral language in dialogue), and (3) use dialogue for only one speaker in a paragraph.

These skills are classified under the dominant discourse mode (i.e., Storytelling, Describing, Informing/Explaining, Persuading) or format type (i.e., Letters, Practical Forms, and Poetry) of the product for which they are first presented. Additional instruction on a skill appears in subsequent grades for the same or other discourse modes or format types.

The classification by discourse modes and formal types parallels the classifications of the Products (see Attachment A). The fact that no skills are listed under a mode/format for any grade level does not mean that component skills are not important at this level, but that new aspects are developed for skills formerly introduced.

Attachment C presents this list of skills by grade level and discourse mode or format type. Attachment D presents these same skills by skill type (e.g., Punctuation) and grade level in order to provide another perspective on the scope and sequence of skills taught in the proposed composition program.

PROGRAM MATERIALS

Three components comprise the instructional materials at each grade level. One component is a student workbook that presents brief instruction and provides appropriate practice. Some of the instructional activities are completed in the workbook. However, students complete the composing tasks on paper distributed by the classroom teacher. Thus students can draft, revise, and edit extensively without being limited to the number of blank pages provided in a workbook, and they can compose their first, revised, and edited drafts on different colors of paper (see Procedures section for a discussion of this important feature).

Another component is the teacher's edition of the student workbook. The teacher's version includes the numbered pages of the student workbook, along with appropriate answers for any practice items. The teacher's version also contains (1) guidelines for presenting instruction, (2) directions for implementing the special techniques that foster the composing process, and (3) suggestions for further class and individual activities.

The third component is an assessment booklet. It contains scoring keys and guidelines for evaluating the compositions that students write during instruction. Scoring these compositions is optional. The booklet also contains the writing prompts, scoring keys, and scoring guidelines that are used for formal assessment. For a discussion of the assessment method and an example of the scoring, see Attachment E. A Class Record Sheet is included in the assessment booklet. The scores students receive on their compositions may be recorded on this sheet for each product completed in instruction. Scores for the corresponding writing sample produced for the assessment prompt are also recorded on the Class Record Sheet.

PROCEDURES

Students prepare for writing through discussion activities led by the teacher. At early grade levels, students practice skills by circling answers to practice items presented in the workbook. As their handwriting skills develop, they complete simple fill-in exercises. Additionally, students read examples of the type of writing they are to undertake. Additional relevant skills of content and form are presented at this time. By grade three, students also prepare for writing by practicing both the syntactic structures presented in sentence-combining exercises and the skills relevant to these structures.

After preparation for writing, students begin planning their compositions, often generating ideas for writing by using a specific set of questions designed to elicit relevant content. Students also take notes they will use during composing. Since elementary school children often plan by drawing and coloring pictures to stimulate their thinking about the content of a task (Gentry, 1980), teachers involve students in this kind of planning as well.

Students then begin drafting their ideas on paper--at least one first draft, one revised draft, and one final, edited draft. These drafts are composed on lined paper, preferably in different colors, and the drafts are kept in students' writing folders. To qualify for publication, students should have at least one draft in each of the different colors assigned to drafting, revising, and editing. They may have, and frequently should have, more than one draft in the designated colors. Using different colors does more than motivate students to become involved in composing as a process; this procedure reinforces the important concept that writing is not a one-time effort: First drafts are not final compositions, and published writers always rework their initial efforts.

Teachers read students' first drafts and suggest possible ways to improve the content and organization. The students then mark up the first drafts and write revised versions. The teacher is again encouraged to review the copy, this time noting surface-level errors. Students correct the errors on this draft and then rewrite the paper in final form on the color designated for finished copy. The composition is ready for evaluation and publication. Although this procedure describes only one draft for revising and editing, students may, as noted above, work through several such "same-color" drafts before they prepare the final copy. Peer critiquing is also undertaken during revising and editing.

The teacher evaluates the composition, using the scoring information provided in the assessment booklet. The teacher may give the student a copy of the filled-in scoring key so that the student is informed about the strengths (and possibly the weaknesses) of his/her composition.

Students' compositions are then published (see earlier section on Publication). The evaluation may be published as well, if both the teacher and the student choose to include it.

Throughout the school year, students and teachers also work with the techniques that foster the composing process (e.g., journal writing, freewriting). Directions for implementing and administering these activities (which can be directed by a classroom aide or tutor) are included in the teacher's materials.

SCHEDULING

Writing instruction has a strong positive influence on students' developmental and educational progress. Instruction and practice in writing has even increased students' scores on tests of reading (Graves, 1981). Consequently, writing should be part of students' daily activities.

The composition program proposed here provides adequate instruction and practice so that written-language activities can be scheduled for 20-30 minutes per day throughout the school year. This much time can easily be arranged for writing because some of the activities can be undertaken independently, whenever they fit into students' daily schedules (e.g., journal writing), and some can be directed by aides or tutors (e.g., freewriting, word associating). Furthermore, students who become actively involved in the writing process will be eager to do some of their drafting, revising, and editing at home.

ATTACHMENT A

PRODUCTS BY DISCOURSE MODE OR FORMAT TYPE

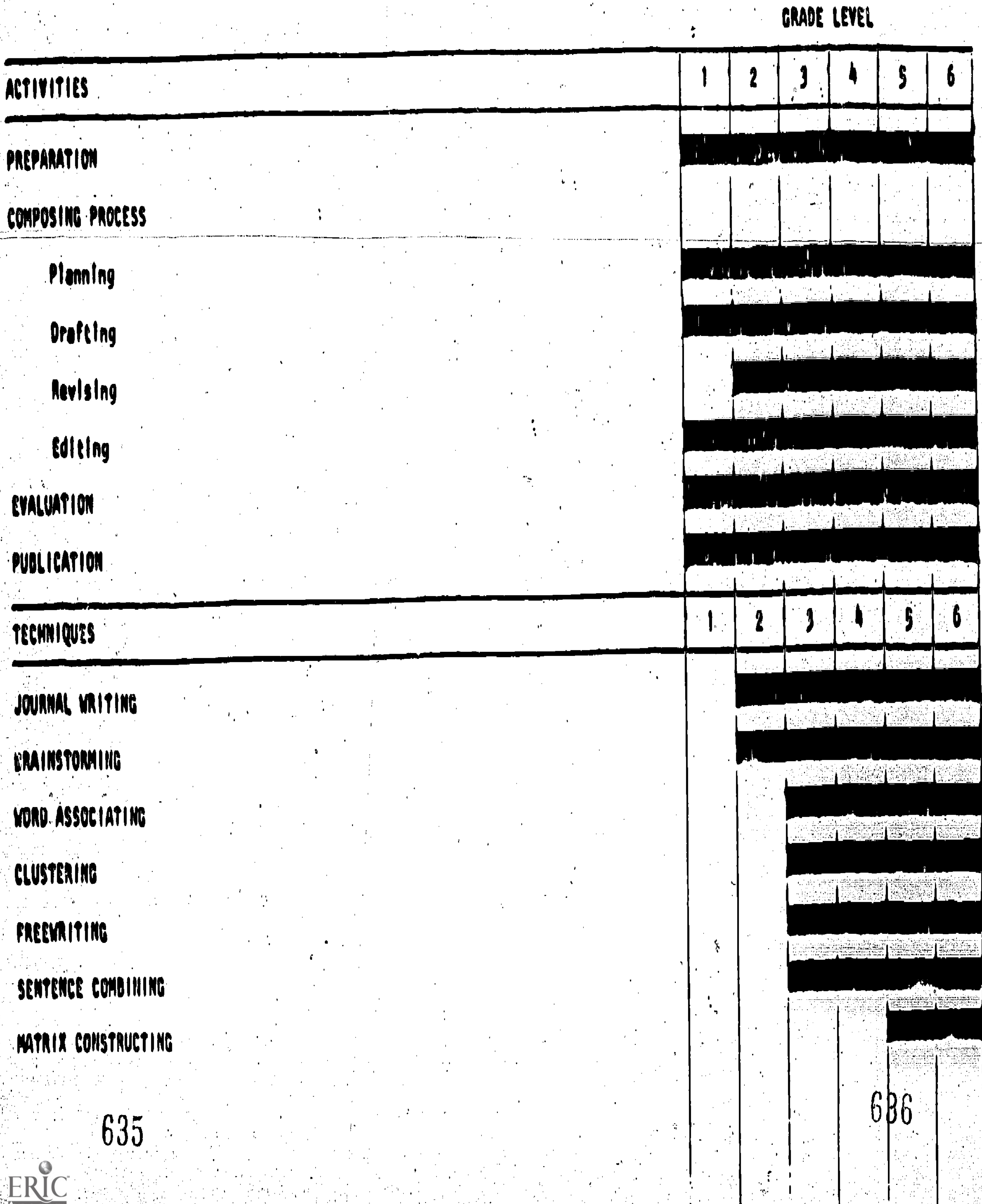
GRADE LEVEL

DISCOURSE MODE	1	2	3	4	5	6
STORYTELLING						
Personal Narratives	■	■	■	■	■	■
Climactic Narratives	■	■	■	■	■	■
DESCRIBING (e.g., object, animal, story character & scene)	■	■	■	■	■	■
INFORMING/EXPLAINING						
Directions (e.g., maps, games)		■	■	■	■	■
Writing in School Subjects (e.g., math word problems; science reports)		■	■	■	■	■
Summaries (e.g., stories, articles, telephone messages)			■	■	■	■
Analyses (e.g., comparison, process)			■	■	■	■
Reports (e.g., book, news, research)			■	■	■	■
PERSUADING (e.g., announcements, letters to editor)			■	■	■	■

FORMAT TYPE	1	2	3	4	5	6
LETTERS						
Personal (e.g., invitations, postcards)			■	■	■	■
Business					■	■
PRACTICAL FORMS (e.g., applications)				■	■	■
POETRY (e.g., cinquain, limericks)			■	■	■	■

ATTACHMENT B

CONTENT OF ACTIVITIES AND TECHNIQUES



17

635

636

ATTACHMENT C

SCOPE AND SEQUENCE OF SKILLS

GRADE LEVEL

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION*	1	2	3	4	5	6
STORYTELLING						
Adds an ending to a story.	X					
Proofreads for omitted and extraneous words.	X					
Constructs a simple sentence.	X					
Uses declarative sentences.	X					
Uses period to end a declarative sentence.	X					
Uses interrogative sentences.	X					
Uses a question mark to end an interrogative sentence.	X					
Uses correct subject-verb and subject-verb-object order.	X					
Capitalizes the first word in a sentence.	X					
Composes a story beginning.		X				
Identifies events appropriate for a fictional narrative.		X				
Identifies the problem and solution in a story.		X				
Identifies events appropriate for a personal narrative.		X				
Indents first word of paragraph.		X				
Uses exclamatory sentences.		X				
Uses an exclamation point to end an exclamatory sentence.		X				

*Additional instruction on skills may appear in subsequent grades and in other discourse modes or format types.

Attachment C (cont.)

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION (cont.)	GRADE LEVEL 1	2	3	4	5	6
STORYTELLING (cont.)						
Capitalizes the names of persons.		X				
Capitalizes the personal pronoun "I."		X				
Writes story titles.			X			
Uses chronological ordering.			X			
Uses chronological transitional expressions.			X			
Uses synonyms.			X			
Locates words in a dictionary.			X			
Uses sentences with compound predicate elements.			X			
Uses transitive and intransitive verbs appropriately.			X			
Uses regular and irregular verbs appropriately.			X			
Uses comma to separate items in a series.			X			
Capitalizes days of the week and holidays.			X			
Capitalizes a one-word title.			X			
Uses consistent verb tense.				X		
Uses playscript format.				X		
Uses dialog for only one speaker in a paragraph.				X		
Expands sentences with phrases.				X		
... sentences with prepositional phrases.				X		

Attachment C (cont.)

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION (cont.)	GRADE LEVEL 1	2	3	4	5	6
STORYTELLING (cont.)						
Uses nominative and objective pronouns appropriately.				X		
Uses apostrophe in contractions.				X		
Uses comma after a noun of direct address.				X		
Capitalizes first word in dialogue quotations.				X		
Uses quotations marks, comma, and terminal punctuation in a dialogue quotation.				X		
Capitalizes first, last, and important words of a title.				X		
Uses a variety of viewpoints.					X	
Constructs a table of contents.					X	
Constructs a complex sentence.					X	
Expands sentences with clauses.					X	
Uses adverbial clauses of time.					X	
Uses past and past-participle forms of verbs appropriately.					X	
DESCRIBING						
Identifies appropriate descriptive attributes.	X					
Uses sensory imagery.		X				
Uses specific terms.		X				

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION (cont.)	1	2	3	4	5	6
DESCRIBING (cont.)						
Writes compound sentences.			X			
Uses adjectives appropriately.			X			
Uses adverbs appropriately.			X			
Expands sentences with words.			X			
Uses correct subject-verb-object-phrase order.			X			
Uses a comma in a compound sentence.			X			
Uses similes.				X		
Uses spatial ordering.				X		
Uses spatial transitional expressions.				X		
INFORMING/EXPLAINING						
Identifies procedural steps.		X				
Uses imperative sentences.			X			
Uses a period to end an imperative sentence.			X			
Uses sentences with compound subjects.			X			
Uses sentences in which the subject and verb agree (subject before verb).			X			
Uses *a* and *an* appropriately.			X			

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION (cont.)	GRADE LEVEL 1	2	3	4	5	6
INFORMING/EXPLAINING (cont.)						
Limits a topic when given a general subject.				X		
Uses logical ordering.				X		
Uses consistent viewpoint.				X		
Uses paragraphing.				X		
Uses the relative pronouns <u>who</u> and <u>that</u>.				X		
Uses adjective clauses.				X		
Uses object clauses.				X		
Constructs a topic sentence.					X	
Constructs supporting statements for a topic sentence.					X	
Uses a lead paragraph.					X	
Limits a paragraph to one main idea.					X	
Writes a headline.					X	
Distinguishes between fact and opinion.					X	
Uses concise sentences.					X	
Constructs a story-line outline.					X	
Uses antonyms.					X	
Locates words in reference sources.					X	
Uses logical transitional expressions.*					X	

...kind of transitional expression that is logical for the content/context will vary both for informative ..., subsequently, persuasive-writing modes.

SKILLS BY DISCOURSE MODE OF FIRST PRESENTATION (cont.)	GRADE LEVEL 1	2	3	4	5	6
INFORMING/EXPLAINING (cont.)						
Uses consistent verb tense.					X	
Uses interrogative pronouns.					X	
Uses the relative pronoun <u>whose</u>.					X	
Uses comma after introductory transitional expressions.					X	
Uses apostrophe in plural possessive forms.					X	
Writes an introductory paragraph for a report.						X
Uses quotations in a report.						X
Paraphrases sources.						X
Uses library reference sources.						X
Takes notes from a reference source.						X
Constructs a topic outline.						X
Constructs footnotes.						X
Constructs a bibliography.						X
Uses sentences in which the subject and verb agree (verb before subject).						X
Uses the relative pronoun <u>whom</u>.						X
Uses subject clauses.						X

: C (cont.)

DISCOURSE MODE IN ... NTATION (cont.)	1	2	3	4	5	6
:						
:ifies a reward for behavior as an audience motivator.				X		
announcement format.				X		
:ifies an issue and an opinion.					X	
:ifies reasons that support a point of view.					X	
:s a point of view.					X	
cause and effect structure in sentences with adverbial clauses of ason.					X	
clauses subordinated by a conjunction.					X	
a comma after introductory adverbial clauses.					X	
:ifies evidence that supports reasons.						X
participial phrases.						X

FORMAT TYP... ...RST PRESENTATION	1	2	3	4	5	6
ses the parts of a personal letter (heading, greeting, body, closing, nature).			X			
:s the parts of a personal letter in the correct space.			X			
udes all elements of address and return address when addressing an velope (name, street address, city, state, zip code).			X			
:s the address and return address in the correct space on an envelope.			X			

24

C (cont.)

FORMAT TYPE OF FIRST PRESENTATION (cont.)	1	2	3	4	5	6
ont.)						
uates the greeting and closing of a personal letter.			X			
a comma between city and state.			X			
a comma in the date.						
alizes the greeting and closing of a personal letter.			X			
alizes personal titles (e.g., Mr., Ms., Dr.).			X			
alizes names of streets, cities, states.			X			
es the message and address in the correct space on a postcard.				X		
talizes names of rivers, lakes, buildings.				X		
apostrophe in singular possessives.				X		
appropriate formal language in a business letter.					X	
oses the parts of a business letter (heading, inside address, greeting, dy, closing, signature).					X	
es the parts of a business letter in the correct space.					X	
tuates the greeting and closing of a business letter.					X	
italizes the greeting and closing of a business letter.					X	
italizes proper adjectives and names of organizations.					X	
ls in blanks of forms with personal information.				X		
es periods at the ends of abbreviations.				X		

652

nt C (cont.)

Y FORMAT TYPE OF FIRST PRESENTATION (cont.)	1	2	3	4	5	6
ont.)						
s periods at the end of initials.				X		
italizes initials in the names of persons.				X		
italizes abbreviations for days, months, streets.				X		
italizes first letter in names of countries.				X		
italizes first letter in non-personal proper names.					X	
s connotative language.				X		
s words that convey emotion.				X		
s verse format.				X		
ctuates poetry.				X		
italizes first line in verse.				X		
s metaphors.					X	
s alliteration.					X	
s onomatopoeia.					X	
s rhyme.					X	
s couplets.					X	
s rhythm.						X
s meter.						X

Above grade columns: GRADE LEVEL

ATTACHMENT D

SKILLS BY CATEGORY AND GRADE LEVEL

	GRADE LEVEL 1	2	3	4	5	6
riod to end a declarative sentence.	X					
question mark to end an interrogative sentence.	X					
exclamation point to end an exclamatory sentence.		X				
period to end an imperative sentence.			X			
comma in a compound sentence.			X			
mma to separate items in a series.			X			
comma between city and state.			X			
comma in the date.			X			
tes the greeting and closing of a personal letter.			X			
riods at the ends of abbreviations.				X		
riods at the end of initials.				X		
mma after a noun of direct address.				X		
otation marks, comma, and terminal punctuation in a dialogue tion.				X		
ostrophe in singular possessives.				X		
strophe in contractions.				X		
tes poetry.				X		

Additional instruction on skills may appear in subsequent grades.

(cont.)

	GRADE LEVEL					
	1	2	3	4	5	6
cont.)						
ma after introductory transitional expressions.					X	
strophe in plural possessive forms.					X	
omma after introductory adverbial clauses.					X	
es the greeting and closing of a business letter.					X	
N						
zes the first word in a sentence.	X					
zes the personal pronoun "I."		X				
zes the names of persons.		X				
zes days of the week and holidays.			X			
zes a one-word title.			X			
zes personal titles (e.g., Mr., Ms., Dr.).			X			
zes the names of streets, cities, states.			X			
izes the greeting and closing of a personal letter.			X			
izes first, last, and important words of a title.				X		
izes first word in dialogue quotations.				X		
izes initials in the names of persons.				X		
izes names of countries.				X		
izes names of rivers, lakes, buildings.				X		
izes abbreviations for days, months, streets.				X		
izes first line in verse.				X		

(cont.)

)	GRADE LEVEL 1	2	3	4	5	6
ION (cont.)						
lizes first letter in non-personal proper names.					X	
lizes names of organizations and proper adjectives.					X	
lizes the greeting and closing of a business letter.					X	
s first word of paragraph.		X				
es the parts of a personal letter (heading, greeting, body, ing, signature).			X			
the parts of a personal letter in the correct space.			X			
es all elements of address and return address when addressing nvelope (name, street address, city, state, zip code).			X			
the address and return address in the correct space on an envelope.			X			
layscript format.				X		
ialogue for only one speaker in a paragraph.				X		
aragraphing.				X		
nnouncement format.				X		
the message and address in the correct space on a postcard.				X		
in blanks of forms with personal information.				X		
erse format.				X		
es the parts of a business letter (heading, inside address, greeting, , closing, signature).					X	

(cont.)

GRADE LEVEL

	1	2	3	4	5	6
.)						
ne parts of a business letter in the correct space.					X	
ts a story-line outline.					X	
ts a table of contents.					X	
ts a topic outline.						X
tations in a report.						X
nsitive and intransitive verbs appropriately.			X			
ular and irregular verbs appropriately.			X			
ectives appropriately.			X			
erbs appropriately.			X			
icles a and an appropriately.			X			
tences in which the subject and verb agree (subject before verb).			X			
ective clauses.				X		
ect clauses.				X		
sistent verb tense.				X		
inative and objective pronouns appropriately.				X		
e relative pronouns who and that.				X		
e relative pronoun whose.					X	

D (cont.)

)	1	2	3	4	5	6
)						
nterrogative pronouns.					X	
ast and past-participle forms of verbs appropriately.					X	
onsistent verb tense.					X	
dverbial clauses of time.					X	
lauses subordinated by a conjunction.					X	
ubject clauses.						X
articipial phrases.						X
he relative pronoun <u>whom</u>.						X
ntences in which the subject and verb agree (verb before subject).						X
RUCTURE						
ucts a simple sentence.	X					
eclarative sentences.	X					
nterrogative sentences.	X					
orrect subject-verb and subject-verb-object order.	X					
xclamatory sentences.		X				
ucts a compound sentence.			X			
mperative sentences.			X			
entences with compound subjects.			X			

	GRADE LEVEL 1	2	3	4	5	6
CTURE (cont.)						
tences with compound predicate elements.			X			
rect subject-verb-object-phrase order.			X			
sentences with words.			X			
sentences with phrases.				X		
tences with prepositional phrases.				X		
ts a complex sentence.					X	
sentences with clauses.					X	
se and effect structure in sentences with adverbial clauses son.					X	
cific terms.		X				
onyms.			X			
notative language.				X		
ds that convey emotion.				X		
iles.				X		
onyms.					X	
ropriate formal language in a business letter.					X	
aphors.					X	

(cont.)

GRADE LEVEL

	1	2	3	4	5	6
t.)						
lteration.					X	
matopoeia.					X	
me.					X	
plets.					X	
thm.					X	
ter.					X	
ds for omitted and extraneous words.	X					
ronological ordering.			X			
ronological transitional expressions.			X			
atial ordering.				X		
atial transitional expressions.				X		
gical ordering.				X		
nsistent viewpoint.				X		
a topic when given a general subject.				X		
gical transitional expressions.					X	
ncise sentences.					X	
a paragraph to one main idea.					X	

.)	1	2	3	4	5	6
	GRADE LEVEL					
ILLS						
s words in a dictionary.			X			
s words in reference sources.					X	
rases sources.						X
ibrary reference sources.						X
notes from a reference source.						X
ucts footnotes.						X
ucts a bibliography.						X
ending to a story.	X					
ies events appropriate for a fictional narrative.		X				
ies the problem and solution in a story.		X				
ies events appropriate for a personal narrative.		X				
s a story beginning.		X				
story titles.			X			
variety of viewpoints.					X	

D (cont.)

t.)	1	2	3	4	5	6
ifies appropriate descriptive attributes.	X					
sensory imagery.		X				
EXPLAINING						
ifies procedural steps.		X				
tructs a topic sentence.					X	
tructs supporting statements for a topic sentence.					X	
a lead paragraph.					X	
es a headline.					X	
inguishes between fact and opinion.					X	
es an introductory paragraph for a report.						X
G						
tifies a reward for behavior as an audience motivator.				X		
es a point of view.					X	
tifies reasons that support a point of view.					X	
tifies an issue and an opinion.					X	
tifies evidence that supports reasons.						X

GRADE LEVEL spans columns 1–6.

ATTACHMENT E

EVALUATING WRITING

The method for evaluating writing in this proposed composition program (1) is easy to use without formal training, (2) gives scores specific content features of the writing task, and (3) provides diagnostic information. It combines the simplicity of analytic scales and holistic scoring with the specificity of Primary Trait Scoring (PTS).* For formal assessment, a highly structured prompt is provided. This prompt also facilitates easy scoring because it is always accompanied by scoring criteria that pertain specifically to that prompt. For an example of a prompt, see Figure 1. For the corresponding scoring key, see Figure 2.

As illustrated by the scoring key in Figure 1, the skills array of the scoring matrix consists of the features of good writing that a composition exhibits when a student employs the content and form skills appropriate for both the specific writing task and the grade level of assessment. For example, an explanatory composition that gives directions should present specific instructions sequentially (Mullis, 1975). Also, this information should be conveyed in precise language. The SWRL scoring key displayed in Figure 2 typically simplifies evaluating these characteristics because it describes how they are implemented specifically for this individual task:

*For an explanation of analytic and holistic scoring, see Cooper (1977) and Humes (1980). For an explanation of Primary Trait Scoring, see Klaus et al. (1979) and Humes (1980).

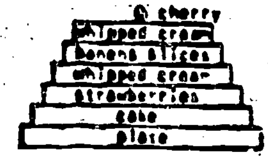
Directions: Read the contents and study the layers of the Fruit and Whipped Cream Treat.

Fruit and Whipped Cream Treat

Contents

1 yellow cake
1 cup of strawberries
1 cup of cut-up bananas
2 cups of whipped cream
1 cherry

Specifications for picture to be drawn:



Write a paragraph that explains to someone who has never fixed food how to put the Fruit and Whipped Cream Treat together.

- Start with the first thing you do.
- Tell about each part of the Treat that you put together.
- Tell about the parts in the order that you must put them together.
- Be sure to use exact words.
- Use words like next to join some of the sentences in your paragraph.

Figure 1. SWRL Writing Prompt.

Scoring Criteria	Good	Acceptable	Unacceptable
CONTENT:			
Includes all parts of the Treat.	___	___	___
Writes about the steps in the order in which they are performed.	___	___	___
Uses specific terms.	___	___	___
Uses logical transitions.	___	___	___
Uses sentences that pertain to the main idea.	___	___	___
FORM:			
Uses correct grammar.	___	___	___
Uses good sentence structure.	___	___	___
Capitalizes correctly.	___	___	___
Punctuates correctly.	___	___	___
Spells correctly.	___	___	___
Writes legibly, with appropriate margins and indentation.	___	___	___

Figure 2. SWRL Scoring Key for Prompt in Figure 1.

Trait	Implementation
Specific information:	Includes all parts of the treat.
	Uses sentences that pertain to the main idea (this criterion also assesses coherence).
Sequential presentation:	Writes about the steps in the order in which they are performed.
	Uses logical transitions (this criterion also assesses coherence).
Precise language:	Uses specific terms.

The SWRL model lists content and form features for separate scoring, and performance ratings of good, acceptable, or unacceptable are given on each criterion. Thus the scoring elicits considerable diagnostic information about the individually listed component skills. It also provides a total score for the writing sample when numerical equivalents are assigned to the good/acceptable/unacceptable ratings of the content and form skills and these individual scores are tallied.

A scoring guide (see Figure 3) accompanies the prompt; this guide describes the features that constitute a good, acceptable, or unacceptable score on each criterion in the scoring key. These guidelines are simple to interpret and thus to use because they correspond to precise individual features that comprise the whole composition.

Scoring is further simplified by a range that spans three broad rating categories. Although authorities differ over the number of ratings a scoring range should span, they generally agree that the fewer the possible rankings, the simpler the scoring. The three-way discriminations evaluators must make do not entail so much information processing

Includes all parts of the Treat (i.e., plate, cake, strawberries, whipped cream, banana slices, whipped cream, cherry).

Good: All parts are included.

Acceptable: Most parts are included.

Unacceptable: Many parts are missing.

Writes about the steps in the order in which they are performed (i.e., cake on plate, strawberries on cake, whipped cream on strawberries, banana slices on whipped cream, whipped cream on banana slices, cherry on whipped cream/cherry on top).

Good: Most or all steps are in the right order.

Acceptable: Many steps are in the right order.

Unacceptable: Many steps are not in the right order.

Comment: If fewer than three steps are included, do not score this criterion.

Uses specific terms.

Good: Specific terms are generally used to explain the procedure; e.g., an exact word like "spread" is used to explain adding a food part to the top of a previous food part.

Acceptable: General terms are used to explain the procedure; e.g., "put on" is used to explain adding a food part to the top of a previous food part.

Unacceptable: Language used does not convey a sense of the procedure.

Uses logical transitions.

Good: Several logical transitions for time (e.g., "then," "next"), space (e.g., "on top of"), and possibly reason (e.g., "because") are used.

Acceptable: Few logical transitions are used.

Unacceptable: No logical transitions are used.

Figure 3. Sample of SWRL Scoring Guidelines.

as do systems with wider ranges: Wider ranges require evaluators to remember too many distinctions for the different ratings of all the features to be scored.

Field testing of this scoring model has produced data on its scoring ease (Humes, 1979). Scorers were given no prior training, yet results suggest that scoring of writing samples with this system is rapid and yields consistent rank-order scores. The mean time for reading and scoring papers at grade 2 was only 1.95 minutes, with a range of 1.17 to 2.67 across scorers; the mean time for the process for grade 6 papers was 2.8 minutes, with a range of 1.67 to 3.33. Although individual evaluators scored papers consistently higher or lower than did other scorers, the rank order of samples generally matched that of other evaluators.

Another study (Cronnell, 1980), provided evidence of the reliability of the method. Twenty-two classroom teachers untrained in the SWRL scoring method scored 511 writing samples. These same samples were also scored by professionals highly familiar with the method. When teachers' and professionals' scorings were compared, a high level of agreement was evidenced, with correlations of over .80 for one-third of the classes.

The careful structuring of the complete assessment package (i.e., prompt, scoring key, and scoring guidelines) is an exacting and time-consuming task. However, great effort put into assessment development results in minimal effort for users. Evaluating writing is then a comparatively simple task that can be quickly expedited by the classroom teacher.

References

Beaven, M. H. Individualized goal setting, self-evaluation, and peer evaluation. In Cooper, C. R., & Odell, L. (eds.), Evaluating writing: Describing, measuring, judging. Urbana, Ill.: National Council of Teachers of English, 1977, pp. 135-156.

Bereiter, C. Development in writing. In R. W. Tyler & S. H. White (chair), Testing, teaching, and learning. Washington, D.C.: NIE 1979.

Braddock, R., Lloyd-Jones, R., & Schoer, L. Research in written composition. Champaign, Ill.: National Council of Teachers of English, 1963.

Britton, J., Burgess, T., Martin, N., McLeod, A., & Rosen, H. The development of writing abilities (11-18). London: MacMillan Education Ltd., 1975.

Cooper, C. R. Holistic evaluation of writing. In Cooper, C. R. & Odell, L. (eds.), Evaluating writing: Describing, measuring, judging. Buffalo: State University of New York, 1977, pp. 3-31.

Cronnell, B. The scoring of writing samples: A study (Technical Note No. 2-81-03). Los Alamitos, Calif.: SWRL Educational Research and Development, 1980.

Educational Testing Service. How the essay in the CEEB English composition test is scored: An introduction to the reading for readers. Princeton: Educational Testing Service, 1976.

Elbow, P. Writing without teachers. New York: Oxford University Press, 1973.

Gentry, L. A. A new look at young writers: The writing-process research of Donald Graves (Technical Note No. 2-80-07). Los Alamitos, Calif.: SWRL Educational Research and Development, 1980.

Graves, D. Balance the basics: Let them write. New York: Ford Foundation, February, 1978.

Graves, D. H. Where have all the teachers gone? Language Arts, Vol. 58, No. 4., April, 1981, pp. 493-4.

Humes, A. A method for evaluating writing samples. (Technical Note No. 2-80-02). Los Alamitos, Calif.: SWRL Educational Research and Development, 1980a.

Humes, A. Writing sample, scoring keys, and scoring guides: A pilot study (Technical Note No. 3-79-05). Los Alamitos, Calif.: SWRL Educational Research and Development, 1979.

Humes, A., with Cronnell, B., Lawlor, J., & Gentry, L. (eds.), Moving between practice and research in writing. Los Alamitos, Calif.: SWRL Educational Research and Development, 1981.

Kinneavy, J. L. A theory of discourse: The aims of discourse. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1971.

Klaus, C. H., Lloyd-Jones, R., Brown, R., Littlefair, W., Mullis, I., Miller, D., & Verity, D. Composing childhood experience: An approach to writing and learning in the elementary grades. St. Louis: CEMREL, Inc., 1979.

Lloyd-Jones, R. Primary trait scoring. In Cooper, C. R. & Odell, L. (eds.), Evaluating writing: Describing, measuring, judging. Buffalo: State University of New York, 1977, pp. 33-66.

Moffett, J. Teaching the universe of discourse. Boston: Houghton Mifflin Company, 1968.

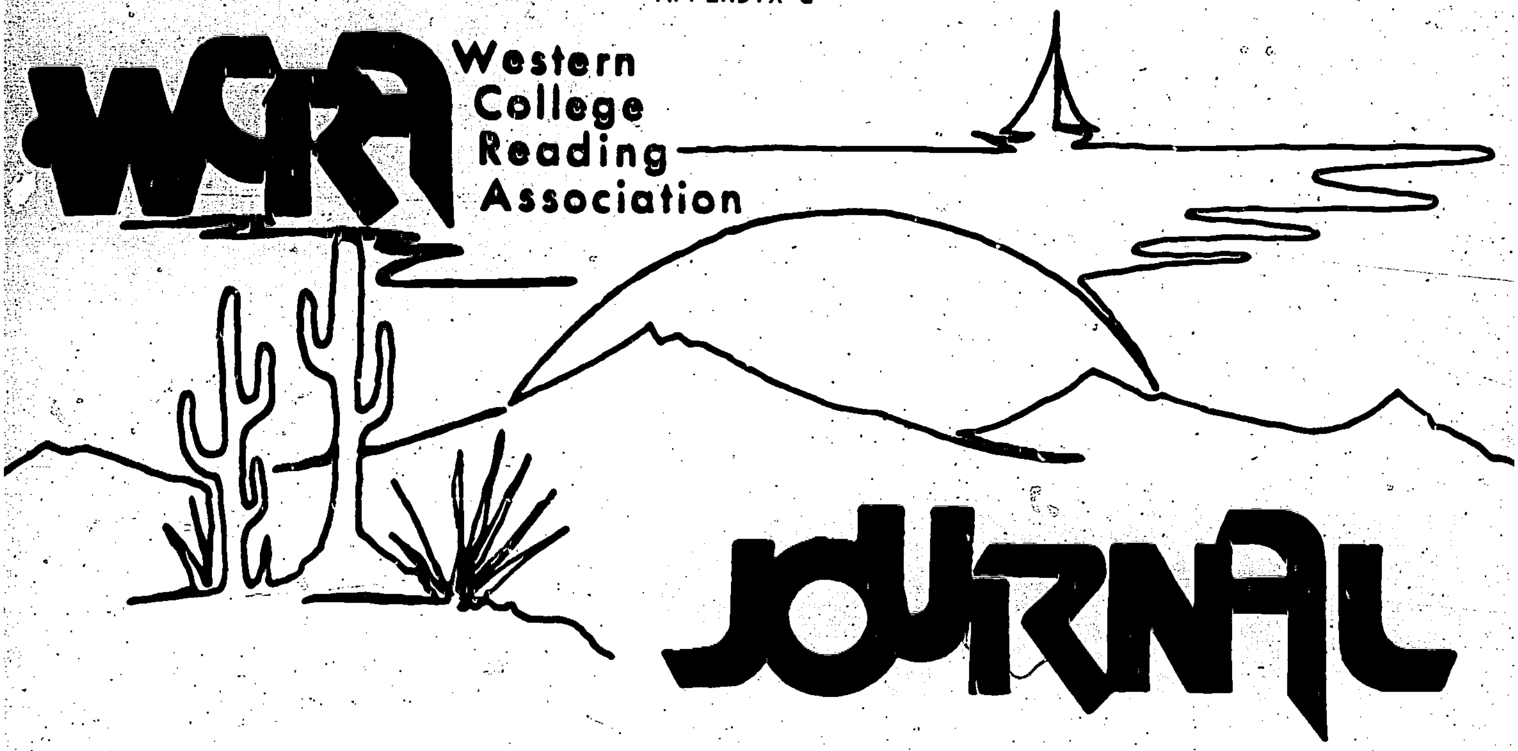
Moffett, J. & Wagner, B. J. A study centered language arts curriculum (second edition). Boston: Houghton Mifflin Company, 1976.

Mullis, I. V. S. The primary trait system for scoring writing tasks. Denver: National Assessment of Educational Progress, 1975.

National Assessment of Educational Progress. Writing achievement, 1969-70: Results from the third national writing assessment. Denver: Education Commission of the United States, 1980.

Shaughnessy, M. P. Errors and expectations: A guide for the teacher of basic writing. New York: Oxford University Press, 1977.

APPENDIX G



Volume 2 | Number 1 | Fall 1982

*** Contents ***

In this issue of *WCRA Journal* appear articles by Bruce Cronnell and Ann Humes on the use of word processors for writing skills development, by Carol Bogue on problem-solving, and by Willas Sayre on readability of vocational texts. The editors hope that you will find these articles both interesting and stimulating.

We continue to solicit your contributions to the *Journal*. We can only bring you high quality articles if you take time to write them and then submit your manuscripts to us. If you want a copy of submission guidelines, or if you want to submit a manuscript, send to: Michael F. O'Hear, Coordinator, Transitional Studies, Indiana University-Purdue University, 2101 Coliseum Blvd. East, Fort Wayne, Indiana 46805.

Computers, Word Processors, and Composition Instruction

Bruce Cronnell
Ann Humes
SWRL Educational Research and Development

When some people hear about using computers in composition instruction, they react with a question: "Using computers in composition instruction?" Many people reason that an activity like writing cannot be combined with a machine such as a computer. However, growing evidence suggests that writing and computers can, in fact, be combined. (See Lawlor, 1982, for a review of some work relating computers and composition.)

First of all, it's important to remember that the computer is only a tool — humans operate it. Humans can't write anything without tools. Originally people used stone and clay, then paper and pen, then pencils and yellow pads. Fountain pens were an advance over pens that had to be continually dipped in ink. Now most people use ballpoint or felt-tipped pens. Surely we all appreciate these changes and don't want to return to quill pens or clay tablets.

The most recent advance involves the use of word processors that are tied into computers. A word processor has a keyboard very similar to that of a typewriter, but the text shows up on a screen and is stored in the computer. Word processors are especially valuable because of what can be done with the text after it has been typed (or while it is being typed). Text can be deleted — from a single letter up to a page or more. And when something has been deleted, the text closes up so that the deletion is not evident. Similarly, any amount of text can be inserted, and the word processor accommodates the added text. Text can also be rearranged; for instance, with a few simple operations, a paragraph can be moved anywhere in the text (even into the middle of another paragraph). With a word processor, other minor but bothersome corrections can be made, such as changing capitals to lower-case (and vice versa) and reversing letters that have been transposed. Then after the changes have been made, the final text can be printed. And if this final text isn't what is desired, the writer can go back and change some more. In other words, the word processor is a great machine for revising.*

Considerable research (reviewed in Gentry, 1980) has indicated that revision is one of the most important parts of the composing process. But research has also indicated that students don't receive very much instruction on the revising process and that they don't revise very much (Applebee, 1981; Bamberg, 1978; Hoetker & Brossell, 1979; Murray, 1978).

*The word processor can do many other things that are also useful, but are probably of more interest to someone preparing formal documents. For example, it can automatically place footnotes and number pages when it prints. It can easily make all kinds of changes in margin width. And it can change spacing with a few quick operations. Although all of these features — and many more — are really quite marvelous, they're not critical for composing, which is the focus of this paper.

One reason that students don't revise is that revision means a great deal of work. Marking up a paper is easy enough — crossing off, drawing lines to move pieces, inserting new information. But recopying the whole paper is a time-consuming, tedious, and unrewarding task. And when we recopy, we often make mistakes that have to be fixed up, leaving a product that still doesn't look good.

With a word processor, the work of revision is much easier. Major and minor changes can be made without recopying. The changes take place right before the writer's eyes, and the clean, revised text can be read immediately. In fact, with a word processor, revision can be fun. Consequently, word processors should be valuable in teaching and encouraging students to revise.

Since revision is one of the most valuable uses of word processors, it's an area that needs attention. Several things can be done instructionally with revision; a few ideas are outlined below.

1. After receiving appropriate instruction on revision, students could be given prewritten text and asked to revise it on the word processor. The computer could respond to student changes in various ways. For example, it could refuse to let the student go on if a change were not made. Or if the student did not make the right change, the computer could make the change for the student and point it out.

2. After a student has composed a text, the computer could suggest that he/she make revisions. For example, when two short sentences are found together, the computer could ask the student whether the sentences might be combined. Or when a very long sentence is found, the computer could ask whether the sentence might be split. When the same noun or noun phrase occurs in two consecutive sentences, the computer could ask the student whether the second one might be replaced with a pronoun. When a paragraph is very short, the computer could ask the student whether he/she wishes to add more information. Of course, the student wouldn't be required to make the changes suggested by the computer, but the approach is valuable because the computer interacts with the student while she/he is writing or revising. All writers could be helped by having a personal editor; the computer can help be an editor for the student writer. (Some computer-editing programs are currently being developed; see Frase et al., 1981).

3. After a student has composed a text, the computer could look for specific kinds of errors in the text (e.g., spelling), mark the line/sentence/paragraph where the error occurred, and suggest that the student correct it.

4. The teacher could also act as editor, interacting with student writing in ways similar to those suggested above.

In our own work, one of the first things we have done with the revision process is to teach sentence combining using a computer. Research (reviewed in Lawlor, 1980) indicates that sentence combining can help increase syntactic fluency and improve student writing. Sentence combining can also be valuable for the revision process (Humes, 1980). We have prepared instruction for sentence combining as a way to begin our study of composition instruction with a computer. We picked sentence combining because it is based on a solid research foundation and because constraints can be built into it that make it amenable to computer-based instruction. For example, the expected output can be specified so that the computer knows what kind of response to look for. Moreover, because most students have had little experience with computers and word processors, this seems to be a straightforward place to begin: Two or more sentences are presented, and the student types in the appropriately combined sentence.

We discovered right away that programming even for such "simple" content as sentence combining is far from easy. Once we begin working with natural language, programming becomes very complex. Moreover, since we are permitting students to type in their own responses, the number of possible responses is very large. (Of course, we assume that students will seriously attempt to get the appropriate answer, but even so, mistakes are possible.) We believe that working with sentence combining is the most appropriate way for us and for the students to begin — it goes beyond simple multiple-choice activities, but stops short of activities that require more complex programming and student-computer interaction.

The computer can be valuable for another area of composition instruction — teaching students to generate content. Depending on the type of discourse the students are to write, the computer can ask heuristic questions about the subject. The students' answers can be stored for later use when the students are actually writing the desired composition. This procedure not only helps students generate content for a specific composing task, but also teaches them to use heuristic probes while composing. (Programs using heuristics are described in Burns & Culp, 1980; Schwartz, 1982.)

The computer can also be valuable for teaching students to arrange content. Since word processors are capable of rearranging text, computer-based instruction could teach students the arranging process and demonstrate the effects of various arrangements. The computer could then work with students as they attempt to arrange content for writing.

A more difficult instructional task has students use a word processor when they actually write. One possibility is to present the student with a simple figure or pattern to describe (for example, an arrangement of geometric shapes). The computer could construct the figure/pattern as the student composed the description. If the student's description were appropriate, the original and the computer-generated figure would match. If they didn't match, the student could be helped to determine where his/her description was inappropriate.

This article has suggested a number of ways in which computers with word-processing capabilities may be useful for composition instruction. Many more possibilities exist — many that haven't even been thought of yet. Computers and word processors are important tools for business, industry, academia, and individuals; we also believe that they can be valuable tools for composition instruction.

References

1. Applebee, A.N. *Writing in the secondary school* (Research Report No. 21). Urbana, IL: National Council of Teachers of English, 1981.
2. Bamberg, B. Composition instruction does make a difference: A comparison of college freshmen in regular and remedial English courses. *Research in the Teaching of English*, 1978, *12*, 47-59.
3. Burns, H. L., & Culp, G. H. Stimulating invention in English composition through computer-assisted instruction. *Educational Technology*, August 1980, 5-10.
4. Frase, L. T., Macdonald, N. H., Gingrich, P. S., Keenan, S. A., & Collymore, J. L. Computer aids for text assessment and writing instruction. *NSPI Journal*, November 1981, *20*, 21-24.
5. Gentry, L. *Textual revision: A review of the research* (Technical Note No. 2-80/11). Los Alamitos, CA: SWRL Educational Research and Development, 1980. (ERIC Reproduction Service No. ED 192 355).
6. Hoetker, J., & Brossell, G. Who (if anyone) is teaching them writing and how? *English Journal*, October 1979, *68*, 19-25.
7. Humes, A. *Specifications for writing instruction* (Technical Note No. 2-80/21). Los Alamitos, CA: SWRL Educational Research and Development, 1980. (ERIC Reproduction Service No. ED 192 376).
8. Lawlor, J. (Ed.) *Computers in composition instruction*. Los Alamitos, CA: SWRL Educational Research and Development, 1982.
9. Lawlor, J. *Improving student writing through sentence combining: A literature review* (Technical Note No. 2-80/05). Los Alamitos, CA: SWRL Educational Research and Development, 1980. (ERIC Reproduction Service No. ED 192 356).
10. Murray, D. M. Internal revision: A process of discovery. In C. R. Cooper & L. Odell (Eds.), *Research on composing*. Urbana, IL: National Council of Teachers of English, 1978.
11. Schwartz, H. J. Monsters and mentors: Computer applications for humanistic education. *College English*, 1982, *44*, 141-152.

Readability and Vocational Texts: A Model for Support Services

Willas Sayre
University of California-Irvine

For many years textbook readability has been a subject of importance to those concerned with education at the elementary and secondary levels. California state law, for example, requires all books to have the results of a readability formula presented before they can be considered for adoption. Readability has gained in significance in higher education with the development of open admissions policies. Community colleges serving vocational students with a wide range of reading abilities are faced with a choice concerning readability and student reading skills. They may limit admission to vocational courses with difficult texts to those students whose reading level is high enough to master the material, or they may select simplified texts that reach students with weaker skills. Although both of these options have some usefulness, a third option focuses on making the more difficult text accessible to the student reading at a level lower than that of the text.

At Long Beach City College, instructors in the vocational subjects and the vocational education and reading instructor have worked on a coordination of efforts to provide a maximum amount of support for students. Individual assistance is provided in the Center for Learning Assistance Services, where students come on their own time and do assignments to raise their skill levels. A separate reading course that differs somewhat from other reading courses given at the college is also offered for vocational students. Where appropriate, reading workshops are provided in the classes. What is more important, readability surveys of textbooks and supplementary readings are completed for vocational subjects along with assessment of the reading skills of students in the classes. By providing both of these services, the vocational reading instructor can offer excellent support for classes in which the reading level falls below that of the required text.

To measure readability, several formulas have been developed. These are generally based on word length and sentence length, although some use lists of commonly occurring words as well. All methods provide an estimate of the relative difficulty of material in terms of levels roughly equivalent to grade levels. A text may have sections with different levels, and an average readability ranging from 1 or 2 through 16+ or CG (college graduate). Using more than one formula on the same selection yields a fairly close correspondence of results. Therefore, for practical purposes, the simplest and most direct methods are the most desirable. Two that have been used in the vocational program at Long Beach City College are the SMOG(3) and the Flesch(1).

SMOG grading uses words of three or more syllables as the most important gauge, although sentence length is also important. In using this formula at least three passages are selected, one at the beginning, one in the middle, and one at the