DOCUMENT RESUME

ED 255 180 IR 011 549

AUTHOR Herman, Joan; And Others
TITLE Microcr"puters in Schools: Toward More Effective Pract. .. CSE Monograph.
INSTITUTION Califo.. .a Univ., Los Angeles. Center for the Study of Evaluation.
SPONS AGENCY National 'nst. of Education (ED), Washington, DC.
PUB DATE Nov 84
GRANT NIE-G-84-0012-P4
NOTE 225p.; Research into Practice Project.
PUB TYPE Collected Works - Conference Proceedings (021) -- Guides - Non-Classroom Use (055) -- Viewpoints (120)

EDRS PRICE MF01/PC09 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; Computer Literacy; *Computers; *Computer Science Education; *Computer Software; Elementary Secondary Education; Evaluation Criteria; *Instructional Design; Instructional Materials; Literature Reviews; School Districts
IDENTIFIERS Research into Practice Project; *Software Evaluation

ABSTRACT

The seven sections in this volume analyze ways to strengthen the educational impact of microcomputers in the schools and to maximize their potential as important learning and instruction tools. The following papers are included: (1) "Human Cognition and the Use of New Technologies" (Richard Mayer); (2) "Don't Bother Me with Instructional Design: I'm Busy Programming!" (M. David Merrill); (3) "From Domain-Referenced Curriculum Evaluation to Selection of Educational Microcomputer Software" (Wells Hively); (4) "Comprehensive Evaluation of Computer Courseware: Getting Back to the Basics" (Kenneth A. Sirotnik); (5) "Computers in the Classroom: Another Case of the More Things Change the More They Stay the Same?" (Jeannie Oakes and Mark Schneider); (6) "The School District Role in Introducing Microcomputers: A Contingency Approach" (Richard C. Williams, Adrianne Bank, and Carol Thomas); and (7) "Some New (and Old) Directions for Computer Courseware" (J. D. Fletcher). The introduction lists eight references, and additional references are listed for each paper. (LMM)

DELIVERABLE - NOVEMBER 1984

RESEARCH INTO PRACTICE PROJECT

CSE Monograph

Microcomputers in Schools: Toward More Effective Practices

Joan Herman

Project Director

Grant Number

NIE-G-84-0112, P4

Center for the Study of Evaluation

Graduate School of Education

University of California, Los Angeles

The project presented or reported herein was supported pursuant to a grant from the National Institute of Education, Department of Education. However, the opinions expressed herein do not necessarily reflect the position or policy of the National Institute of Education and no official endorsement by the National Institute of Education should be inferred.

Table of Contents

INTRODUCTION

Computers have arrived in force in the nation's classrooms, at a rate that defies accurate estimation. The numbers multiply faster than surveys can be conducted and reports written. A few examples attest to this growth. Studies by the National Center for Educational Statistics report that there were 31000 microcomputers in schools in the fall of 1980, and by 1983 that figure had increased tenfold (Bell, 1984). Documenting similar patterns, a recent comprehensive study by the Johns Hopkins Center for Social Organization of Schools reports that the number of elementary schools having at least one microcomputer jumped from less than one percent in July, 1980 to 51 percent in January, 1983. And while estimates by the United States Office of Technology Assessment indicated that the percentage of districts using microcomputers was likely to jump from 50 to 90 percent from 1982 to 1984, a survey conducted by the National School Boards Association in April, 1984 found that 96 percent of their school sample was using microcomputers for instructional purposes.

Enthusiasts of this technology surge have spoken glowingly of the computer's revolutionary power and efficiency, of its ability to individualize and manage instruction, of the value of its word and other data processing capacities, of students' and teachers' needs to be on the cutting edge of these promising developments. And the technology does have a variety of appealing features which create a strong potential for enhanced student learning. Many have pointed to the value of the computer's capacity for giving immediate feedback and for providing extensive individualized interaction with students

(Komoski, 1984), capacities which far transcend its print and more traditional counterparts. Others have identified additional features which distinguish computer learning from other learning environments. For example, Linn and Fisher (1984) suggest six key features which give the computer exceptional power for facilitating cognitive skills acquisition: 1) interactiveness, reflecting the cyclical process of providing information, demanding active student response, and giving rapid feedback; 2) preciseness, reflecting the need for specificity and completeness in communicating with the machine (e.g., in programming); 3) consistency, reflecting the computer's capacity to provide uniform instruction and feedback; 4) challenge and motivational value, reflecting the intrinsic motivation and reward value of interacting with the machine and with stimulating software; 5) complexity, reflecting the many functions which the machine can handle and its ability to develop complex mental models; and 6) provisions for multiple divergent responses, enabling it to reinforce creativity and problem solving by accepting many rather than a single correct answer.

What is the impact of this relatively new technology on the ools? How is it being used and to what extent does its use capitalize on the computer's unique features? The Johns Hopkins' study found that approximately 40 percent of all instructional time on microcomputers in elementary schools is spent using courseware to practice mathematics, language facts, spelling and various other memorization tasks, while 35 percent is spent in computer programming activities and the remaining 25 percent is occupied playing games (Johns Hopkins, 1983). Mirroring these findings, examinations of available software indicate that half are of the drill and practice type; the second largest category is tutorial with 19 percent of the entries, followed by educational games (12%) and simulations with only five percent (Bialo and Erickson, 1984).

The instructional uses of the computer at the secondary level vary significantly from those at the elementary school level. Here, fully two-thirds of the time spent on computers is devoted to programming and computer literacy activities; drill and practice occupies about 20 percent, and the remainder is split among games, word processing, and business applications. However, while high school students apparently spend considerably more time learning about the computer, their instructional time on the computer in other subjects bears striking resemblance to that at the elementary school: over 60 percent of this "other" time is spent in drill and practice activity. This preponderance of drill and practice at both levels raises obvious questions about the extent to which the power and potential the technology is being utilized in current classroom practice. Several contributors to this volume further pursue these issues.

The reliance on drill and practice in instructional applications raises additional questions about the use of computers in schools. There is evidence to suggest, including the data cited above, that the instructional uses of computers fall predominantly into two broad categories: as a tool for promoting computer literacy and computer programming skills and as a tool for providing remedial instruction, instruction which is likely to involve repetitive drill and practice routines. When one looks at the types of students who are most likely to be involved in each of these kinds of activities, issues of equity inevitably emerge. Not only do disadvantaged and minority students have less access to computers than their more advantaged peers, but their access is more likely to be limited to remedial academic work. Likewise, they are less likely to participate in experiences which will enable them to understand and utilize technologies of the future (Shavelson et al, 1984). It appears, then, that schools may be contributing to a widening gap in technological competence, a gap between more and less advantaged students and between males and females (Linn, 1984). The access of handicapped students to computer-assisted education is another problem (NSBA, 1984).

Despite these problems, however school practitioners feel quite optimistic about the impact of the computer on their schools. While a great majority admit that computers have not changed the methods or content of instruction in their schools, they report having "good" to "excellent" success in providing individualized instruction (NSBA, 1984). Educators responding to a survey of school districts in Ohio, for example, reported that computers have had beneficial effects on student achievements in basic skills and in comprehension of concepts and have enabled more effective remediation of student deficiencies. However, they felt that improved student motivation was the greatest benefit derived from computer use (Morgan, 1984).

While practitioners identify benefits derived from the use of computers in schools, they also acknowledge significant problems and impedements. In the Ohio study, administrators cited "too few hardware units" as the most important problem, followed by lack of adequately trained staff, lack of understanding of the capabilities of the technology and lack of software; the latter difficiency was likewise noted in results from the NSBA Survey.

A recent study of available software by the Educational Products Information Exchange (EPIE) provides additional insight into the software problem (Bealo and Erichson, 1984). Their results are discouraging:

> Development evidence: there was an overwhelming lack of evidence of field testing in the course of program development. Over 80.4% provided no development documentation.
>
> Learner objectives: only one-third of the programs had well-defined, educationally appropriate objectives; more than one half had either no objectives stated, or unclear developmentally inappropriate ones.
>
> Clarity: Clarity as evidenced in directions, frame formatting and content expression was mixed across courseware. In reading, most (55%) did not meet even minimal expectations.
>
> Support materials: Almost 70% included no support materials of any kind. When support materials were provided, they generally were not judged as useful or appropriate.
>
> Instructional documentation: A majority (62%) included either no or inadequate instructions, suggestions or information to aid in integrating a program into the curriculum.
>
> Feedback: While in most cases (69%), feedback was immediate and included some form of reinforcement, less than 20% provided any remediation.
>
> Approach enhances content: More than half failed to use an approach that lent itself to an effective delivery or appealing presentation that clarified or enhanced content.
>
> Tests: The great majority (78%) did not include any test or assessment, and 60% had no form of evaluation.
>
> Branching: Almost 3/4 didn't provide any branching or individualized options.
>
> Management: Most (80%) provided no management system.

The conclusion one must draw from such findings is that much of current educational software is poorly designed and does not utilize research-based principles of instructional design. Nor does it effectively utilize the power of available microcomputer technology.

The papers in this volume analyze ways to strengthen the educational impact of microcomputers in the schools and to maximize their potential as important learning and instruction tools.

Richard Mayer, in "Human cognition and the use of new technologies", questions whether the computers being introduced in schools today will "soon join their teaching machine predecessors, collecting dust in schoolhouse basements.... [will they] become just another costly fad in education?" He answers that their success will depend not only on the power of the technology but also on the educational effectiveness of the instructional materials they incorporate. He argues specifically that the effective use of the technology is tied to the educational value and pedagogic usefulness of current theories of human learning and cognition. Mayer suggests insights and principles derived from cognitive psychology that can help guide the development of effective computer programs to help students acquire semantic, procedural, and strategic knowledge.

While Mayer describes how recent research in cognitive theory can advance software design, M. David Merrill suggests how principles of instructional design can lead to more effective programs. His paper, "Don't Bother Me with Instructional Design: I'm Busy Programming," provides suggestions for more effective software in three areas: instructional design; display techniques; and human factors. Related to instructional strategies, Merrill considers how to arrange the instructional content and other elements to facilitate learning, e.g., providing generalizations with examples, using attention focusing devices, using response modes which promote active mental processing. With regard to display techniques, he considers ways in which information can be exhibited on the monitor that will enhance students' ability to interact with the materials. Finally, he considers within human factors those characteristics which make software easier and more efficient to use.

While Merrill's and Mayer's principles might suggest potential criteria for evaluating instructional software, Wells Hively reminds us that not all software is intended to serve the same purpose and any valid evaluation system must take account the characteristics of the program and the ways in which it is intended to be used. His paper, "From Domain-Referenced Curriculum Evaluation to Selection of Educational Microcomputer Software" proposes a preliminary scheme for classifying various types of instructional software. The classes of programs he identifies include domains of practice; tutorial; educational games; intrinsic games; intuition building programs; simulation programs; information retrieval programs; and tools and displays. Hively concludes with an important observation: "... overwhelming impression from watching children work with all these different types of programs is that their effectiveness depends at least as much on the classroom context in which they are used as on the properties of the programs themselves." He emphasizes the central role of the teacher in this context and recommends we make it as easy as possible for teachers to integrate software into their curricula and lesson plans.

Ken Sirotnik, too, in his paper "Comprehensive Evaluation of Computer Courseware: Getting Back to the Basics" suggests that we have been too superficial in our approach to evaluating computer technology. Sirotnik maintains that we are "rather easily seduced by the promise of educational innovations ... and in our attempt to evaluate the promise, our thoughts turn more to surface-level, technological issues rather than to deeper meanings that we have known for some time to constitute the basic questions of curriculum and instruction". He points out that courseware does not exist independently of the curriculum, and that any comprehensive evaluation must consider all relevant elements of the teaching and learning experience, and their interaction with educational values and commonplaces.

Jeannie Oakes and Mark Schneider consider the microcomputer's potential for changing the fundamental character of schooling in "Computers in the Classroom: Another Case of the More Things Change the More They Stay the Same." They point out that the potential for change inherent in microcomputers may be enormous, but if we look closely at what actually happens in computerized schooling it bears striking similarity to the status quo of traditional classroom practic Oakes and Schneider consider the factors which are related to ch conclude that the failures of educational change have not been due to the quality of the ideas but because of an inappropriate perspective on how change occurs in schools. They suggest that instead of an RD&D perspective, a culturally responsive perspective be adopted that concentrates on creating self-renewing schools, ones in which the staff works together to identify problems and develop alternatives. They identify local ownership and district support as critical variables supporting innovation and change.

Williams, Bank, and Thomas also emphasize the important role of the district in implementing new technologies in schools. In their paper, "The District Role in Introducing Microcomputers: A Contingency Approach," they maintain that districts need a long range strategic plan for introducing computers and for coordinating hardware, software, and training needs. They suggest that such a plan be ongoing, incremental, adaptive, and self-correcting. A contingency approach is advocated which includes four components: 1) conducting a situation audit of external and internal environments; 2) generating support; 3) formulating district-wide policy; and 4) developing an ongoing operational plan.

J.D. Fletcher provides a vision of future potential and capabilities of computers in schools in his paper, "Some New Directions for Computer Courseware." He explores the evolutionary and revolutionary aspects of computer technology and describes emerging developments that promise real and innovative solutions to instructional problems. He considers these new functionalities in the areas of drill and practice, tutorial dialogue and simulation. Fletcher concludes his vision with a challenge: "...computers will help us better perform the business of instruction as we envision it today. They will also broaden our horizons. They will change and expand our ideas about what instruction is and what it must do. Their challenge to us as educators is as serious as their promise. We should rise to the occasion."

References

Bell, T.H. Effective use of computers in schools require coordinated development. T.H.E. Journal. 11(5) 80-83, 1984.

Bialo, E.R. and Erickson, L.B. Microcomputer courseware: Characteristics and design trends. Paper presented at the annual meeting of the American Educational Research Association. New Orleans, LA, 1984.

Johns Hopkins University. School uses of microcomputers, April 1983.

Komoski, P.K. Educational software and other learning materials: A look at the differences. Paper presented at the annual meeting of the American Educational Research Association. New Orleans, LA, 1984.

Linn, M.C. and Fisher, C. Policy issues in computer learning. Paper presented at the annual meeting of the American Educational Research Association. New Orleans, LA, 1984.

Morgan, J.M. The Ohio survey on school district uses of educational technology. Paper presented at the annual meeting of the American Educational Research Association. New Orleans, LA, 1984.

National School Board Association. Survey of school/home computer uses. 1984.

Shavelson, R.J., Winkler, J.D., Stasz, C, Feibel, W., Robyn, A. and Shaha, S. Teaching mathematics and science: Patterns of microcomputer use. Santa Monica, California: Rand, 1984.

Human Cognition and the Use of New Technologies

Richard E. Mayer

Psychology Department

University of California, Santa Barbara

Introduction

Computer technology is invading our nation's schools. However, the ultimate usefulness of this new technology may be viewed with either optimisim or pessimism. In the optimistic view, computers will becomes aides for teachers, providing help in areas such as instruction, problem solving, and evaluation. In the pessimistic view, computers will become an expensive fad and eventually join their predecessors--teaching machines--collecting dust in the basements of schoolhouses across the nation.

This paper argues that the effective use of computer technology in schools requires an understanding of how humans learn and think. The fulfillment of the optimistic scenario of computers depends on their being used in a way that is consistent with what we know about the psychology of human cognitive processes. In order to avoid the pitfalls of the past, and thus to deny the fulfillment of the pessimistic scenario, we must not base the use of computer technology on psychological principles which are inappropriate.

The tremendous influx of computer technology into our nation's schools has been widely reported. In a recent report to school board members, Fortune (1983) points out that more than 100,000 microcomputers and terminals were installed in schools in 1982, and that there will be almost one million microcomputers in schools by

1985. Similarly, a recent report in News (1983) stated:

> As of last spring, by one count, 29,000 schools provided... microcomputers and terminals for 4,711,000 school students. Another study released last fall found that 60 per cent of the nation's school districts use computers for learning and that the number of elementary schools using them had increased 80 percent over the year before. In fact, computers are multiplying too fast to count; experts figure the statistics are obsolete when they are reported.

In California, the Apple Computer Foundation's "Kids Can't Wait" program is providing one computer system for every school in the state., and the state's "Investment in People" program is providing about $10,000,000 for the improvement of education related to "high technology". Fortune (1983, p. 7) summarizes all of the new programs as follows: "One thing is clear: computers in the school are not just a passing fad."

The urgent need to prepare for the role of computers in schools has been widely recognized. For example, a recommendation from Technology in Science Education: The Next Ten Years (National Science Foundation, 1979) states that "there is an urgent, national need to create an educational system that fosters computer literacy in our society." The report points out that "American education is not only missing a great opportunity, it is failing to discharge a crucial responsibility" (Deringer & Molnar, 1982).

As another example, the President's Report on Science and Engineering Education in the 1980's and Beyond (National Science

Foundation, 1980) cites the decline in national productivity and increase in foreign trade competition as rationale for preparing American students to become better educated in the use of computers. The French government has recognized the impending "computerization of society" and has committed France to a national policy of computer education for all students (Nora & Minc, 1980). In addition, state departments of education in this country have begun to propose computer courses as part of the mandated graduation requirements (California State Department of Education, 1982).

A recent conference on National Goals for Computer Literacy in 1985 (Seidel, Anderson & Hunter, 1982) concluded by calling for "the presence of computers for instruction in all schools for all students" and "the availability of a critical mass of high-quality curricula and courseware." In particular, the conference supported the proposition that a computer should be in every classroom from kindergarten through eighth grade; in grades 8 through 12, computers should be available in a laboratory environment for every student."

The National Council of Teachers of Mathematics (1980) has issued similar recommendations in its report An Agenda for Action: Recommendations for Mathematics of the 1980's. One recommendation concerning computers stated: "Mathematics programs should take full advantage of the power of calculators and computers at all grade levels." More specifically, the report states, "All high school students should have work in computer literacy and hands-on use of computers."

Two Scenarios

The foregoing section demonstrates that computer technology has arrived in our schools. Let me try to describe two scenarios for the role of computers in improving our children's education: a pessimistic scenario and an optimistic scenario.

In order to fully appreciate the pessimistic scenario for the future, consider the past history of technology in the schools. In particular, recall the role of teaching machines in education, and the theory of learning and instruction which supported their use.

Teaching machines clattered onto the scene of American education about 25 years ago (Skinner, 1958). In his classic book The Technology of Teaching Skinner (1968, p.22) introduced an early version of a teaching machine:

> The device is a box about the size of a small record player. On the top surface is a window through which a question or problem printed on paper tape may be seen. The child answers the question by moving one or more sliders upon which the digits 0 through 9 are printed. The answer appears in square holes punched in the paper upon which the question is printed. When the answer has been set, the child turns a knob. The operation is as simple as adjusting a television set. If the answer is right, the knob turns freely and can be made to ring a bell...If the answer is wrong, the knob will not turn. When the answer is right, a further turn of the knob engages a clutch which moves the next problem into place in the window.

Some more sophisticated versions of teaching machines involved answer keys instead of knobs, and even allowed the students to write an answer.

From the beginning, the technological development of teaching machines was closely tied to an underlying theory of human learning. The dominant force in psychology at the time was behaviorism. Hence, the principles of learning by reinforcement guided the use of teaching machines. In particular, the primary instructional materials for teaching machines were teaching programs--a series of simple questions, each requiring an overt response from the learner. For example, a program in high school physics began with the following items (Skinner, 1968, p. 45):

> The important parts of a flashlight are the battery and the bulb. When we "turn on" a flashlight, we close a switch which connects the battery with the ____.
>
> When we turn on a flashlight, an electric current flows through the fine wire in the ____ and causes it to grow hot.
>
> When the hot wire glows brightly, we say that it gives off or sends out heat and ____.

For each item, the student fills in the missing word, and then uncovers the corresponding word or phrase. In the above example, the correct answers respectively are: bulb, bulb, and light. As you can see, the instructional materials are based on the idea that learners must make a response, and that the response must be immediately reinforced.

Skinner's arguments for bringing teaching machines into schools are remarkably similar to many current arguments for using computers in schools. For example, Skinner (1968, p.26) notes that new technology will aid rather than replace the teacher: "The changes proposed should free her for the effective exercise of her (teaching)." Similarly, Skinner (1968, p. 27) addresses the issue of cost: "Can we afford to mechanize our schools? The answer is clearly Yes."

In spite of the early enthusiasm of Skinner and many others, teaching machines did not revolutionize education. This failure to "mechanize teaching" motivates the questions: Will the computers being introduced today soon join their teaching machine predecessors, collecting dust in schoolhouse basements? Will computers, like teaching machines, fail to live up to the claims that have been made for them, and instead become just another costly fad in education? Twenty-five years from now, will we look back on Papert's (1980, p. 13) observation that "very powerful kinds of learning" take place with computers in the same way we now smile at Skinner's (1968, p. 28) claim that "the equipment needed (for educational innovation) can easily be provided"?

Proponents of the pessimistic scenario may answer "yes" to these questions. In the pessimistic scenario, computers do not find a home in American schools. Yet, there are several factors which lessen the appeal of the pessimistic scenario. First, the computer technology of today is far more powerful than the teaching machine technology of 25 years ago. Computers are not constrained by having to provide a series of test items; instead; computers allow for storage of massive data bases, graphics and simulations, interactive communication, and so on. Second, the current state of psychology has changed dramatically over the past 25 years. The behaviorist theories of learning, based largely on animal research, have been replaced by cognitive psychology. Cognitive psychology provides implications for the instructional use of computer technology that are very different from earlier behaviorist-inspired instructional materials.

In the optimistic scenario, modern theories of learning and cognition are used in the development of useful instructional materials for computers. For example, cognitive psychologists tend to view learning as the acquisition of knowledge rather than the acquisition of responses. Mayer (1981) has shown how the analytic theories of cognitive psychology have been applied to several kinds of knowledge:

semantic knowledge--factual knowledge about the world, such as rainfall patterns for South America.

procedural knowledge--knowledge about how to carry out some procedure, such as how to compute in long division.

strategic knowledge--knowledge about how to set goals and monitor progress towards solving a problem, such as how to plan the writing of a research paper.

One of the major accomplishments of cognitive psychology has been the development of techniques for facilitating each of these kinds of knowledge within specific domains (Mayer, 1981). These techniques have implications for how to design effective instructional uses of computers. In the remainder of this paper, examples are given of possible uses of computers to enhance acquisition of each type of knowledge.

The Computer as an Aid to Learning Semantic Knowledge

Semantic knowledge refers to a person's factual knowledge about the world. Examples include knowledge about geography, such as how climate and terrain are related to a region's major crops, or the determinants of the amount of rainfall in a region.

Recent research on the psychology of human learning and cognition suggests a different approach to instruction as compared to the behaviorist approach which dominated during the teaching machine revolution. These differences can be summarized as follows:

<u>active understanding versus passive memorization</u>--The cognitive approach views learning as an active process in which the learner searches for meaning in what is presented, rather than a passive process of performing and remembering what the instructor demands.

<u>assimilative versus additive</u>--The cognitive approacn views learning as a process of connecting new information with existing knowledge structures, rather than adding isolated pieces of information to memory.

<u>cognitive structures versus responses</u>--The cognitive approach views the outcome of learning as a coherent body of knowledge (or "mental model") rather than a set of specific responses for specific stimuli.

If meaningful learning of semantic knowledge is an active process of assimilating and reorganizing information, then computers may be used in a way that encourages active exploration. For example, Collins & Stevens (1982) have developed an "intelligent tutor" that uses an inquiry or Socratic method, and that can be used with existing computers. The system is based on the idea that learning about some new domain, such as geography or meteorology, involves the construction of a "mental model" which relates all of the variables in the system.

Based on the observations of good human tutors, Collins (1977) developed rules for how to engage in inquiry teaching. Some of the main rules for how to teach are summarized below:

1. Ask about a known case, such as "Do they grow rice in China?"
2. Ask for any factors, such as "Why can they grow rice in China?"
3. Ask for intermediate factors, such as "Why do monsoons make it possible to grow rice in China?"
4. Ask for prior factors, such as "What do you need to have enough water?"
5. Form a general rule for an insufficient factor, such as "Do you think any place with enough water can grow rice?"
6. Pick a counterexample for an insufficient factor, such as "Why don't they grow rice in Ireland?"
7. Form a general rule for an unnecessary factor, such as "Do you think it is necessary to have heavy rainfall in order to grow rice?"
8. Pick a counterexample for an unnecessary factor, such as "Why do they grow rice in Egypt when they don't have much rainfall?"

Collins and Stevens (1982) have summarized the strategies that an intelligent tutor should use in teaching a student. Some strategies involve selecting a case, and then using counterexamples. An example of this strategy is demonstrated in the following dialogue (Collins & Stevens, 1982, p. 81):

Tutor: Why do they grow rice in Louisiana?

Student: It's a place where there is a lot of water. I think rice requires the ability to selectively flood fields.

Tutor: O.K. Do you think there's a lot of rice in, say, Washington and Oregon?

Collins's and Stevens's tutor requires a lot of specific knowledge (such as knowledge about geography), as well as procedures for asking questions and strategies for organizing the questions.

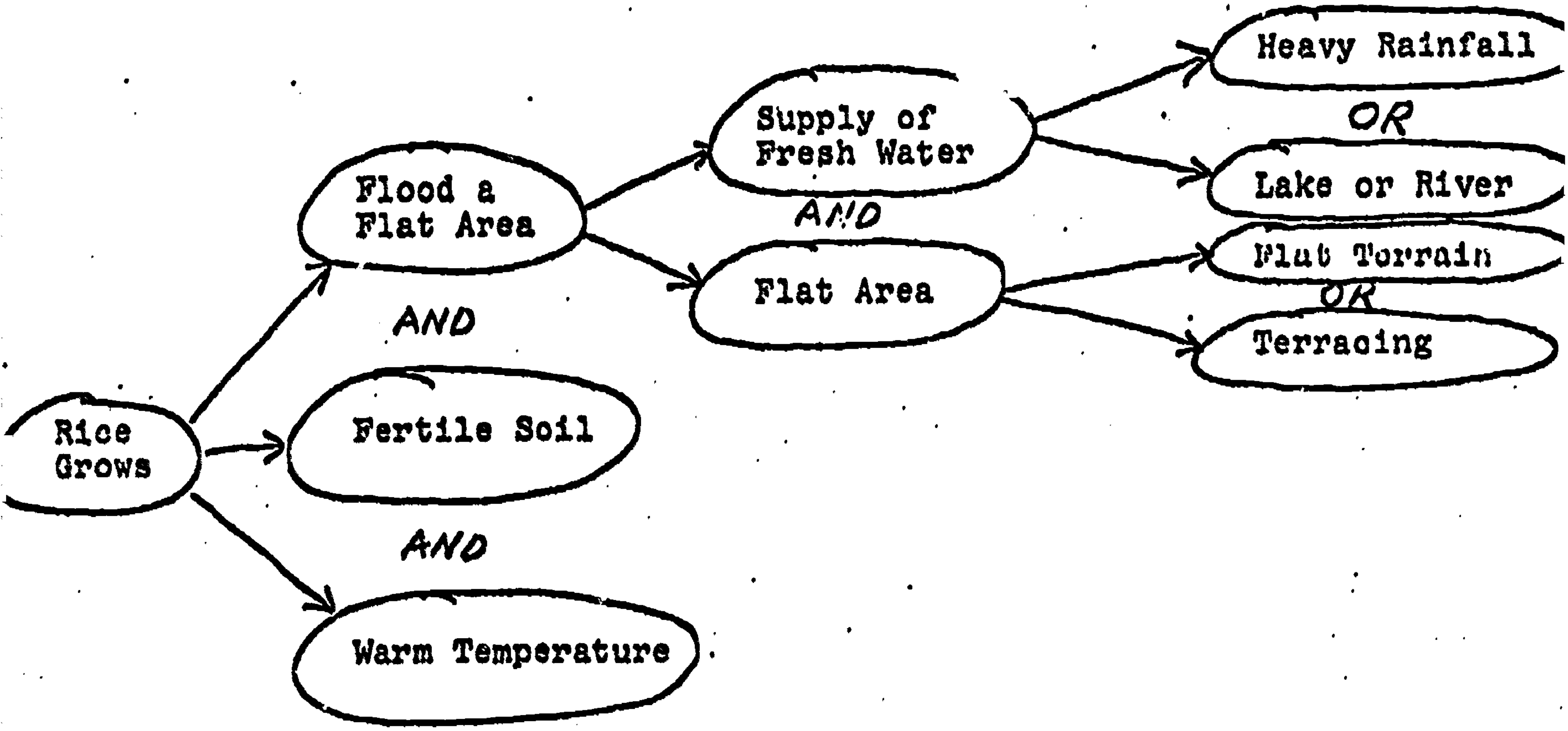
What is learned from a computerized tutor such as the one proposed by Collins and Stevens? A student may form a mental model of the factors involved in growing rice, such as summarized in Figure 1. As you can see, the student builds a coherent structure of factors and relations rather than a set of specific factual answers to specific questions. The mental model allows the student to generate answers to novel questions, and may be used in learning new information.

The use of computers as Socratic tutors represents an exciting possibility, especially in situations where the goal is to teach semantic knowledge. However, the main point in my example is that the way in which the computer is used is determined by the underlying theory of human learning and cognition that is currently available. Thus, the success or failure of computer technology in teaching semantic knowledge depends as much on the educational implications of cognitive psychology as on the power of computer technology itself.

The Computer as an Aid to Learning Procedural Knowledge

Procedural knowledge refers to a person's knowledge about how to

Figure 1. Factors Influencing the Growing of Rice



do something. Examples include knowledge about how to carry out long division or three-digit subtraction. The cognitive approach to procedural knowledge is based on analyzing any procedure into its parts. According to the cognitive approach, the description of procedural knowledge is based on what is learned rather than on how much is learned. Instead of focusing on the percentage of correct answers, the cognitive approach focuses on describing the procedure that the student is using to generate the answers.

Cognitive psychologists have been successful in analyzing many mathematical tasks into their constituent parts. For example, Groen and Parkman (1972) have described several different procedures that children might use to solve problems of the form m + n (where the sum is less than 10). The models are based on the idea that the child uses counting as a way of finding answers to addition problems. Three possible procedures are:

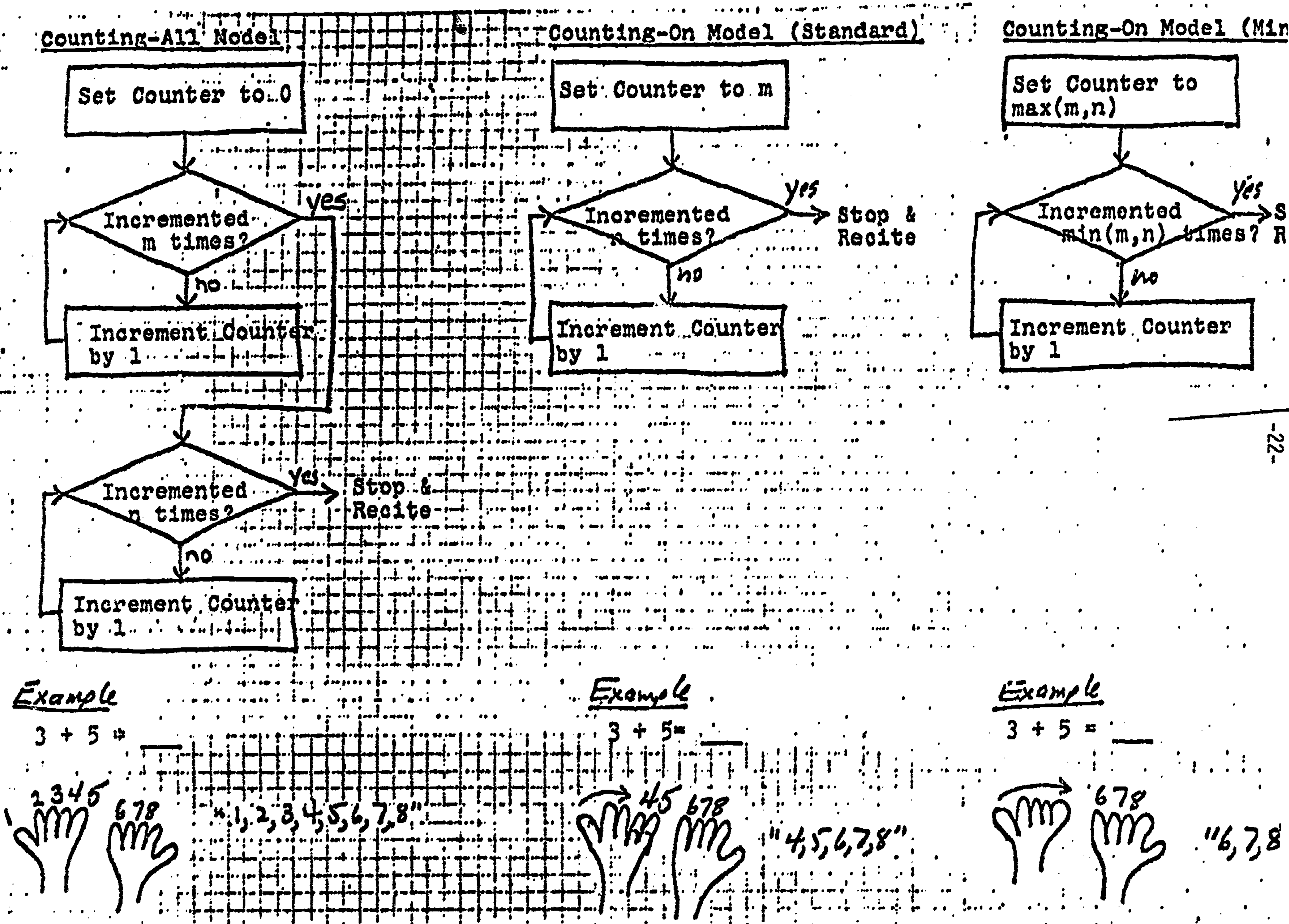
counting-all--Set a counter to 0. Increment it m times and then increment it n times. For 3 + 5, the child recites, "1,2,3...4,5,6,7,8."

counting-on--Set a counter to the first number (m); increment it n times. For 3 + 5, the child states, "4,5,6,7,8."

min model (for counting-on)--Set a counter to the larger of m or n; increment the counter by the smaller of m or n. For 3 + 5, the child states, "6,7,8."

Examples of these three procedures are given in Figure 2; the diamonds represent decisions and the rectangles represent operations. Fuson (1982) has observed a developmental progression in which children move from a counting-all procedure to a counting-on

Figure 2. Three Counting Models of Simple Addition



-22-

BEST COPY

procedure, and eventually to a known-facts procedure in which the answers are memorized.

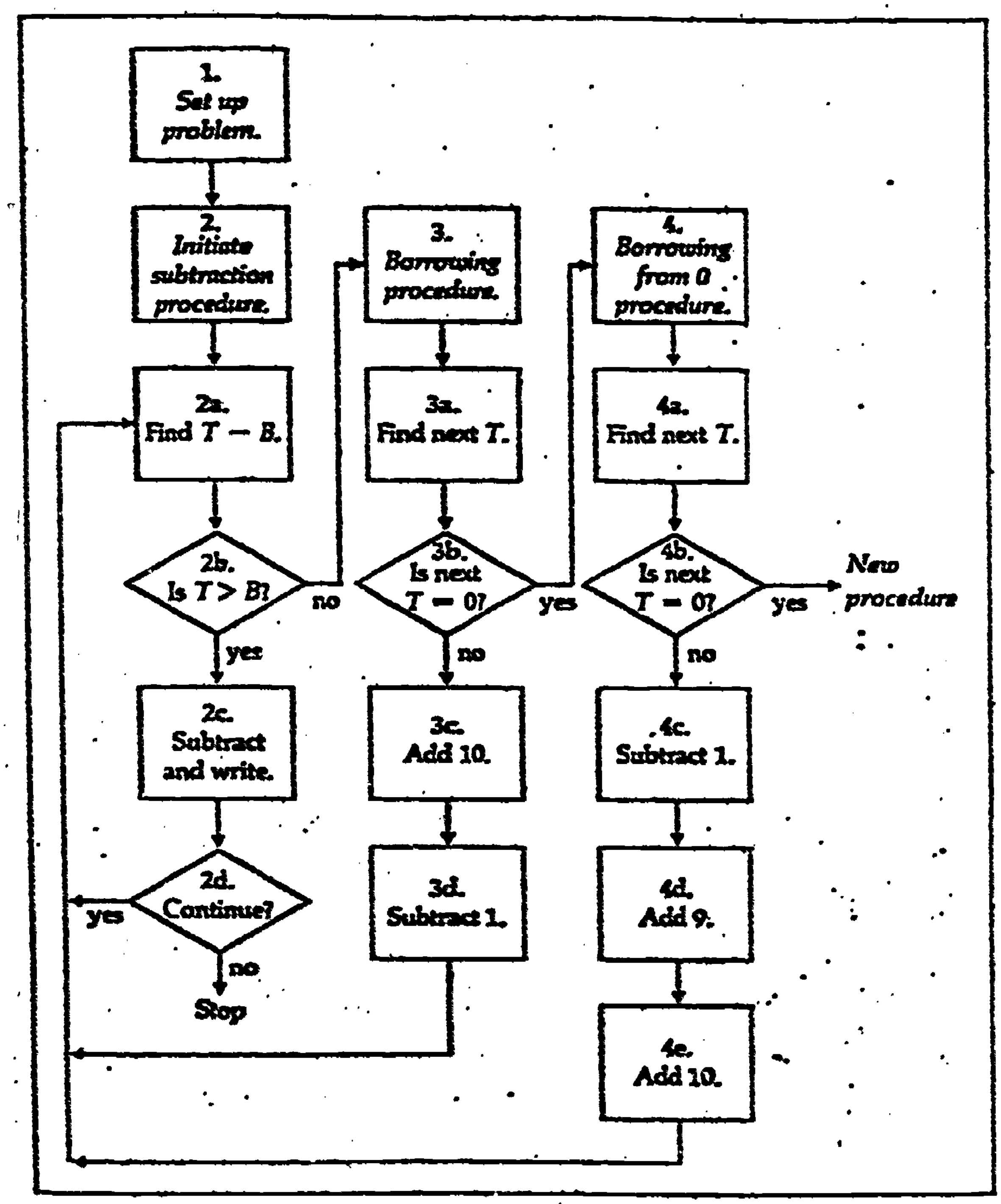
A slightly more complex computational task is three-digit subtraction, such as 697 - 354 =...... Figure 3 shows a computational procedure which generates correct answers for three-digit subtraction problems. If a student possesses this knowledge, then the student will be able to generate correct answers for all three-digit subtraction problems. However, suppose that a student gives answers such as below:

521	819	712	481	655
-418	-203	-531	-380	-160
117	616	221	101	515

We could describe this student's performance by saying that he is right on 40% of the problems. However, a more useful approach is to try to describe the procedure that the student is following. For example, we could say that this student is using the procedure in Figure 3, but with small "bugs"; namely, at steps 2a, 2b, and 2c, the student subtracts the smaller number from the larger number regardless of which is on top.

Brown and Burton (1978) have argued that students' computational performance can be described by saying that they are using a procedure--perhaps with some bugs in it--and applying this procedure consistently to problems. In order to test this idea, Brown and Burton (1978) gave a set of 15 subtraction problems to 1,325 primary school children. Brown and Burton developed a computer program called BUGGY to analyze each student's procedural algorithm for three-digit subtraction. If the student's answers were all correct, BUGGY would

Figure 3. A Process Model for Three Column Subtraction



categorize that subject as using the correct algorithm. If there were errors, BUGGY would attempt to find one bug that could acount for all or most of the errors. If no single bug could account for the errors, then all possible combinations were tried, until BUGGY found combinations that best accounted for the errors. Figure 4 lists some of the most common bugs, such as "borrowing from zero" or subtracting smaller from larger". The BUGGY program was able to describe the performance of about half of the students by providing a list of each student's "bugs". Thus, Brown's and Burton's work provides a means for pinpointing specific bugs in students' computational procedures.

The BUGGY program provides an example of how computer technology can be used to improve the teaching of procedural knowledge. The BUGGY program provides the teacher with a detailed diagnosis of errors in "what is learned" so that the student can be given instruction aimed specifically at remediating the bugs. Again, my point is that the use of computers in teaching of procedural knowledge can be closely guided by existing theories in cognitive psychology.

The Computer as an Aid to Learning Strategic Knowledge

Strategic knowledge refers to knowledge concerning how to set goals, select procedures for achieving goals, and monitor progress toward goals. Examples include knowledge of how to plan the writing of a research paper or how to produce a computer program that accomplishes some task. Research in cognitive psychology emphasizes the role of <u>process</u> rather than <u>product</u> in creative problem solving. For example, consider the following assignments: "Write an essay on whether children should be allowed to choose their own courses in school" or "Write a BASIC program that will take a list of names as

Figure 4. Some Common Subtraction Bugs

Number of Occurrences in 1325 Students	Name	Example	Description
57	Borrow from zero	103 – 45 158	When borrowing from a column whose top digit is 0, the student writes 9, but does not continue borrowing from the column to the left of zero.
54	Smaller from larger	253 –118	The student subtracts the smaller digit in each column from the larger, regardless of which one is on top.
10	Diff 0–N=N	140 – 21	Whenever the top digit in a a column is 0, the student writes the bottom digit as the answer.
34	Diff 0–N=N and move over zero	304 – 75	Whenever the top digit in a column is 0, the student writes the bottom digit as the answer. When the student needs to borrow from a column whose top digit is zero, he skips that column and borrows from the next one.

BEST COPY AVAILABLE

input and give an alphabetized list as output." Instruction could focus on the final product, such as a holistic rating of the final essay or whether the BASIC program runs properly, or could focus on the processes that a person went through in generating the final product, including setting of goals, etc.

Research on the process of writing (Hayes & Flower, 1980) has identified the following processes in writing: planning, in which the author searches memory for ideas and uses these ideas to establish a plan for generating text; translating, the actual production of text; and reviewing, the improvement of the written text. According to these researchers, writing may be viewed as a problem-solving process in which goals are set and monitored.

How can the computer become involved as an aid in writing? One current area is to use the word processing power of computers to stimulate interest in writing and to free children from some of the low level aspects of writing (such as correct spelling, punctuation and penmanship). For example, Sardamalia, Bereiter and Geolman (1982) propose that since the information processing capacity of young writers is limited, and since the mechanical and syntactic aspects of writing are not automatic, emphasis on correctly formed sentences results in poorer overall writing quality. The low level aspects of writing interfere with higher level planning. Evidence for this assertion includes the finding that when children are allowed to dictate their essays (which presumably frees them from some of the low level aspects of writing) they produce longer and higher quality essays as compared to writing.

Currently available word processing systems make revision much easier and free the writer from some aspects of production (such as penmanship and spelling). However, word processors of the future will be even more helpful in stimulating high quality writing. For example, the "Writer's Workbench" (Macdonald, Frase, Gingrich, & Keenan, 1982) is an intelligent computer coach. It consists of a collection of programs which analyze written prose and make suggestions for revisions. The Writer's Workbench is actually in use at Bell Laboratories, with over 1,000 users. You can type your text into the computer, using a standard word processing system. Then, once you have finished your first draft, you can ask the programs from the writer's workbench to suggest revisions in your manuscript.

The writer's workbench consists of three major parts: a proofreader, a style analyzer, and an on-line English reference guide. The proofreader consists of the following programs:

spelling--lists all words that may be misspelled, and allows the user to specify any new words (such as jargon, proper names, and acronyms) to the list of acceptable words.

punctuation--lists cases where punctuation may be needed or where existing punctuation may be incorrect.

double words--lists all cases in which a word is repeated.

faulty phrasing--lists phrases which may not be coherent.

split infinitives--lists all instances of split infinitives.

An example of the output of the proofreading program is shown in Figure 5. As can be seen, the program points out possible errors as well as making suggestions for how to correct the errors.

The style analyzer consists of the following programs:

Figure 5. Output From A Proofreader Program

INPUT: Our report, "The Basic Fundamentals of Computatonal Complexity", is enclosed. Please send any recomended changes at your earliest convenience, thanks.

PROOFR OUTPUT:

************************** SPELLING **************************

Possible spelling errors in examplefile are:

Computatonal recomended

If any of these words are spelled correctly, later type
spelladd word1 word2 ... wordn
to have them added to your spelldict file.

************************** PUNCTUATION **************************

The punctuation in examplefile is first described.

2 double quotes and 0 single quotes
0 apostrophes
0 left parentheses and 0 right ones

The program next prints any sentence that it thinks is incorrectly punctuated and follows it by its correction.

line 1
OLD: Our report, "The Basic Fundamentals of Computatonal Complexity".
NEW: Our report, "The Basic Fundamentals of Computatonal Complexity,"
line 3
OLD: earliest convenience, thanks.
NEW: earliest convenience. Thanks.

For more information about punctuation rules, type:

punctrules

************************** DOUBLE WORDS **************************

For file examplefile:

No double words found

************************** WORD CHOICE **************************

Sentences with possibly wordy or misused phrases are listed next, followed by suggested revisions.

beginning line 1 examplefile
Our report, "The *[Basic Fundamentals]* of Computatonal Complexity", is enclosed.

beginning line 2 examplefile
Please send any recomended changes *[at your earliest convenience]*.

file examplefile: number of lines 3, number of phrases found 2

——————— Table of Substitutions ———————

PHRASE SUBSTITUTION

at your earliest convenience: use "soon" for " at your earliest convenience"
basic fundamentals: use "fundamentals" for " basic fundamentals"

************************** SPLIT INFINITIVES **************************

For file examplefile:

No split infinitives found

style--provides readability indices, measures of average word length and average sentence length, the percentage of verbs in the passive voice, the percentage of nouns that are nominalizations, the number of sentences that begin with expletives, and other such information.

prose--compares the style statistics listed above with some standard measures; if the text's measures are outside of the standards, the program prints an explanation of why the text may be hard to read and prints suggestions for how to correct the problem.

find--locates individual sentences that contain passive verbs, expletives, nominalizations, "to be" verb forms, and other potential problem sentences.

The on-line reference programs include information on the correct use of 300 commonly misused words and phrases, a computerized dictionary, and general information about the writer's workbench. Additional programs rate the words in the text for abstractness-concreteness, rate the paragraph organization, and detect possible instances of sexist language.

Other writer's helper systems include JOURNALISM, a proofreader that comments on the organization and style of news stories (Bishop, 1975), and CRES, a proofreader that identifies uncommon words, long sentences, and difficult phrases in NAVY documents (Kincaid, Aagard, O'Hara, & Cottrell, 1981).

Intelligent computer coaches for writing may help writers to develop more productive writing strategies. For example, in early drafts more attention can be devoted to the organization and goals of the document, since proofreaders will detect lower level errors. In addition, writers are encouraged to engage in more extensive revision cycles, allowing for refinement of writing strategies. Unfortunately, there is very little empirical information concerning the effectiveness of writing coaches, but Macdonald et al. (1982) report that writers tend to like the programs.

Goldstein (1980) has developed a computer coach to teach general problem-solving strategies. For example, a student is asked to play a computer game that requires the use of strategic thinking. Throughout the game, the computer coach makes suggestions or observations about the strategy that the student is using. Goldstein (1980, p. 53) states that "the coach's function is to intervene occasionally in student-generated situations to discuss appropriate skills that might improve the student's play." Thus, an ultimate use of computers may be to expand the power of human strategic thinking. However, as Hayes and Flower (1980) and Goldstein (1980) have pointed out, successful computer coaches must be based on useful theories of human thinking (such as Newell & Simon, 1972). Again, the usefulness of a computer coach is tied to the underlying theory of cognitive processing.

Conclusion

We began with a pessimistic and an optimistic scenario for the role of computers in education. This paper then briefly explored examples of how computers can be used to help learners acquire semantic, procedural, and strategic knowledge. The major theme of this paper has been that the effective use of computer technology in schools is tied to the educational value of current theories of human learning and cognition. Another way to state this theme is to say that the future of computer technology in schools depends on both the technological power of computers and the pedagogic usefulness of cognitive psychology.

A quarter of a century ago, American education was introduced to the technological innovation of teaching machines supported by a behaviorist psychology of learning. Today, schools are again being asked to participate in a technological revolution; however, the technological innovation involves computers, and the dominant psychology of learning is cognitive psychology. The realization of the optimistic scenario depends on our ability to extract what is useful from the cognitive psychology of human learning and cognition and to creatively apply the information to the development of computer-based instructional mateials. Blindly using computers, without making use of what we now know about human learning and cognition, is likely to result in the realization of the pessimistic scenario.

References

Bishop, R.L. (1975). The JOURNALISM programs: Help for the weary writer. _Creative Computing_, _1_, 28-30.

Brown, J.S., & Burton, R.R. (1978). Diagnostic models for procedural bugs in basic mathematical skills. _Cognitive Science_, _2_, 155-192.

California State Department of Education. (1982). _Proposed graduation requirements_. Sacramento, California.

Collins, A. (1977). Processes in acquiring knowledge. In R.C. Anderson, R.J. Spiro, & W.E. Montague (Eds.), _Schooling and the acquisition of knowledge_. Hillsdale, NJ: Erlbaum.

Collins, A., & Stevens, A.L. (1982). Goals and strategies of inquiry teachers. In R. Glaser (Ed.), _Advances in instructional psychology: Vol. 2_. Hillsdale, NJ: Erlbaum.

Deringer, D.K., & Molnar, A. (1982). Key components for a national computer literacy program. In R.J. Seidel, R.E. Anderson, & B. Hunter (Eds.), _Computer literacy_. New York: Academic Press.

Fortune, R.F. (1982). The computer: Its impact on the classroom. _California School Boards_, _42_, (3), 5-6.

Fuson, K.C. (1982). An analysis of the counting-on procedure in addition. In T.P. Carpenter, J.M. Moser, & T.A. Romberg (Eds.), _Addition and subtraction: A cognitive perspective_. Hillsdale, NJ: Erlbaum.

Goldstein, I. (1980). Developing a computational representation for problem solving skills. In D.T. Tuma & F. Reid (Eds.), _Problem solving and education_. Hillsdale, NJ: Erlbaum.

Groen, G., & Parman, J.M. (1972). A chronometric analysis of simple addition. _Psychological Review_, _79_, 329-343.

Hayes, J.R. & Flower, L. (1380). Identifying the organization of writing processes. In L.W. Gregg & E.R. Steinberg (Eds.), Cognitive processes in writing. Hillsdale, NJ: Erlbaum.

Kincaid, J.P., Aagard, J.A., O'Hara, J.W., & Cottrell, L.K. (1981). Computer readability editing system. IEEE Transactions on Profession Communication, PC-24, 38-41.

Macdonald, N.H., Frase, L.T., Gingrich, P.S., & Keenan, S.A. 1982). The writer's workbench: Computer aide for text analysis. Educational Psychologist, 17, 172-179.

Mayer, R.E. (1981). The promise of cognitive psychology. San Francisco: Freeman.

National Council of Teachers of Mathematics. (1980). An agenda for action: Recommendations for mathematics in the 1980's. Reston, Virginia.

National Science Foundation. (1979). Science and engineering education for the 1980's and beyond. Washington, DC.

Newell, A., & Simon, H.A. (1972). Human problem solving. Englewood Cliffs, NJ: Prentice-Hall.

Nora, S., & Minc, A. (1980). The computer of society: A report to the President of France. Cambridge, MA: MIT Press.

Papert, S. (1980). Mindstorms. New York: Basic Books.

Scardamalia, M., Bereiter, C., & Goelman, H. (1982). The role of production factors in writing ability. In M. Nystrand (Ed.), What writers know. New York: Academic Press.

Seidel, R.J., Anderson, R.E., & Hunter, B. (1982). Computer literacy. New York: Academic Press.

Skinner, B.F. (1968). The technology of teaching. New York: Appleton-Century-Crofts.

Skinner, B.F. (1958). Teaching machines. Science, 128, 969-977.

DON'T BOTHER ME WITH INSTRUCTIONAL DESIGN: I'M BUSY PROGRAMMING!

Suggestions for more effective educational software.

M. David Merril
Professor of Educational Psychology and Technology
University of Southern California

A recent comment by Larry Lipsitz, the editor of Educational Technology Magazine, caused me to remove my PASCAL disk from my disk drive, to temporarily put aside my attempts at programming educational software and to boot my word processor and write a few comments about educational program design. "We publish very few instructional design papers anymore;" he said, "I think instructional design is dead!"

"Dead!" I exclaimed. "How can instructional design be dead? Hundreds of people, most of whom have never even heard of instructional design, are at this very moment sitting in front of their personal computers entering computer code to create educational software. Many of these programs have been, are being or will be reproduced and sold to unsuspecting parents, teachers and students. If these self-styled cottage-industry educational technologists are not engaged in instructional design what are they doing?"

Rather than being dead instructional design is undergoing a new birth. The academic approach of formal education is being challenged by creative ideas that are radically different from the past. The explosion of new software promises greater achievement than has ever before been possible.

However, for every effective innovative program that appears there also appear many programs that are innovative only because they are on the computer. These products frequently lack even the fundamentals of effective educational design. In this paper we will explore a few basic techniques of instructional design which are too often ignored by the developers of many of these new educational products.

The earliest educational programs were merely drill and practice and were little more than workbooks put on the computer. In fact some publishers did exactly that, put their already published workbooks on the computer. The result is a very dull program that may provide some skill development if the student already knows the material and if sufficient external reward or punishment can be arranged to keep the student at what otherwise is a very boring activity. Unfortunately many people still equate computer based education with such boring drill and practice programs. Advocates often cite the patience of the computer as a virtue of such programs. Little is said about the patience of the student.

Figure 1 illustrates a short sequence of such a program from a major publisher. There is no instruction prior to what is illustrated. After signing on, the student is presented frame A. The ? symbol is flashing. The student responds by typing in the missing number. If the student is incorrect frame B appears. The incorrect response flashes. After two incorrect tries the student is given the correct answer and presented the next problem. If the student is correct frame C appears. The cute graphic apple with the worm is in color and the worm crawls out of the apple. Pressing the return key gets the next problem. There are only three correct reward graphics. Frame D shows a dragon which breaths fire when the student is correct. The effectiveness of such reinforcers is quickly dissipated and the student is left with the boring task of doing one problem after another with nothing but right and wrong feedback. If the student does not know how to do the problem there is no remedial help, except to call the teacher. Ironically, this program is very expensive!

Too many educational product designers assume that their only responsibility is to provide opportunities for practice. In this author's opinion educational software should do more than merely present a workbook to the student. With the capacity provided by a personal computer courseware should provide not only opportunity for practice but information about how to perform the task as well. Adequate educational software should teach.

In this paper we shall review three classes of instructional design techniques that are necessary if educational courseware is to teach. These techniques can be grouped into those that affect three areas of instructional design: instructional strategy, screen design and human factors. Each of the techniques discussed will be illustrated by two versions of a program on poetic meter. The NOT-SO-GOOD version appears in Figures 2 and 3. This is really a straw man program created esp lly to illustrate what not to do. Perhaps it resembles programs you ve used. The second version (Figures 5 through 11) is a commercial product designed by the author.[1] In this product we have attempted to

1. Introduction to Poetry: Poetic Meter. Two disk program for Apple II or IIe. Includes a teacher editor. Developed by MicroTeacher, Inc. Distributed by Edu-Were Services. 28035 Dorothy Drive, Agoura, CA 91301

use most of the design techniques disscussed. The displays include represent only a few of the displays in the program.

INSTRUCTIONAL STRATEGIES

This section identifies design strategies which are believed to have a direct impact on the amount of learning that occurs as a result of instruction. Instructional design strategy involves the arrangement of content elements and other information in such a way that learning is facilitated. In this section we have attempted to identify a few strategies which have been shown to enhance or inhibit learning.

Page Turning

In the early days of educational television lectures were often put on the TV screen. The result was deadly. Today we have textbooks put on the monitor. Also deadly! If the monitor merely substitutes for a page of text there is little reason to put the material on the computer. Too many educational computer programs present a paragraph of text and then asked the student to press the space-bar or return key to continue. Continuing means to turn the page and see the next paragraph of text.

Many authors and instructors are accustomed to one-way linear presentations such as lectures and textbooks. In such presentations student interaction is seen as an interruption not an essential part of the presentation. As a consequence few designers conceptualize instruction as a conversation or if they do the input from the student seems to be of secondary importance serving only to indicate readiness to continue.

Look at Figure 2. This represents the first few displays of the NOT-SO-GOOD Poetic Meter program. Frames A, B and C are consecutive;

several displays are skipped between frames C and D. This is an example of the text-on-the-screen characteristic of too many educational software packages. A paragraph is presented and then the computer waits until the student presses the return key which results in another paragraph being presented. The material takes no advantage of the dynamic nature of the screen. As can be seen from the illustrations one paragraph scrolls up to make room for the next at the bottom of the screen.

To be effective educational software must be more than text on the monitor. There is no advantage to having students read text from a monitor which they can more easily read from the printed page. In short, avoid merely putting text on the screen; avoid mere page turning.

Generality-rich, Example-poor

Since, as an instructor, we usually understand what we are presenting there is a tendency to present one idea after another with little illustration. Illustrations and examples often seem to slow down the presentation. Text books often contain few illustrations because of the need to cover the subject matter in limited space. Adding unnecessary illustrative material "pads" the book making it unnecessarily lengthy and expensive. As a consequence of this disposition much of the CAI which has been prepared models lectures and textbooks rather than conversations. Ideas are presented one after another with little illustration and little opportunity for student interaction. As a consequence students do not have a chance to digest one idea before the next one comes.

Look at Figure 2. Note that one idea tumbles after another each time the return key is pressed. In frame A the purpose of the lesson is stated but not illustrated. In frame B important prerequisite information is reviewed but not illustrated. In frame C three different ideas, stressed syllables, rhythm and poetic feet are each presented one after another without illustration.

Instruction is more effective if each major idea is represented by a general statement of the idea (generality) followed by examples and finally by the opportunity for the student to demonstrate that the idea is understood (practice). In short, **avoid generality-rich but example-poor presentations. Each idea should be presented by a generality, examples and practice.**

Remember Only

Perhaps the biggest problem in all education is the tendency to tell the student something and then ask the student to repeat what was said rather than applying the idea to a new situation. It is easy to ask a student to repeat what has been learned but much more difficult to figure out how the student should apply the information. Application means that we must find specific cases which can be represented in our instructional materials and then we must figure out what the student does when he or she applies the information that has been presented. Furthermore, all of us have had many examples of remember testing but many fewer examples of application testing. In formal educational settings we are accustomed to recall but not to application.

In Figure 3, frames B and C, we see examples of remember questions. In this displays the student is asked to remember the definitions of each kind of meter. If the student is right the next question is presented; if wrong a message instructs the student to try again. The try again message is repeated until the student stumbles onto the correct answer. Too often this is the only kind of practice that is provided for the student. However, in the NOT-SO-GOOD Poetic Meter program the student is also ask to apply the definitions by identifying the type of meter involved in a series of words (See Figure 3, frame D). A running total indicates how many the student correctly classified.

Too often educational products neglect having the student apply what has been learned. Or if there is application it is insufficient to allow the student to demonstrate what has been learned or for the designer to determine if the student has learned what was taught. Instruction is more effective if the student is required to apply ideas to new situations rather than merely repeating the statement of the idea. In short, **avoid remember only practice.**

Attention Focusing

Not all communication is designed to be instructional. One of the things that characterizes instructional communication is the addition of information that is not part of the subject matter but which is included to help the student understand the subject matter content. Text books use headings, bold face lettering, italics, arrows, exploded drawings and many other devices to facilitate the student's understanding of the content presented. This attention focusing information should serve two primary functions: first, it attempts to relate the illustrations to the generalities being taught and second, it points out the critical characteristics of the example being presented.

Computer displays have a wide variety of display characteristics which can be used for this attention focusing function. All of the characteristics of text presentation are also present on computer displays. For example, words can be underlined, appear in bold face, italic or other type styles. The display can include color which can be used to tie ideas together or emphasize important portions of the display. In addition, however, there are many dynamic characteristics of the computer display that are not available in print. These include: timing where text is put up a little at a time for emphasis, animation where text or figures are moved about to show relationships, flashing or inverse text which can direct attention to a given portion of the screen, sound which can attract attention or show patterns in the presentation.

Too often these characteristics are used for entertainment and have no instructional function. Putting text up letter by letter, word by word or moving across the screen like a sign may be clever but often only serves to make the text harder to read. Having the computer beep when text is entered or displayed often does little more than disturb others in the room. Colorful displays that show rainbow patterns of text or figures too often are only aesthetic and distract from the purpose of the instruction. If these devices are used they should serve an instructional purpose. If they do not they may prove to be more of a distraction than a facilitation.

Look at Figure 1 again. The animated figures used for reinforcement are entertaining the first time they appear but they have no instructional value over that of saying, "You are right." They do not show the relationship between an example and a generality nor do they focus the student's attention on relevant aspects of the examples.

Look at Figure 4. This is a short sequence from another commercial program on fractions.[2] In frame A an animated figure pulls the shading from the first box across the screen to the second box (frames B and C). After moving the shading the figure walks to the bottom of the screen

2. Edu-Ware Fractions. Agoura, CA: Edu-Ware Services, Inc., 1982.

(frame D) to point to the symbolic equivalent of what has been done. this enables the student to visually and dynamically see that 1/2 is equal to 2/4. In this sequence of frames the animation directs the student's attention to relevant characteristics of the display and helps the student understand the relationships involved. The animation is not only relevant but probably shows relationships that would be difficult to explain with words or symbols alone.

Look at Figure 7, frame A. The directions at the top of the display indicate that the student should, "Listen to this passage." The syllables of the poem are then displayed one by one. Each syllable is accompanied by a tone. Stressed syllables are accompanied by a high tone whereas unstressed syllables are accompanied by a low tone. In this way the student can hear the stress pattern in the passage while seeing the passage displayed. Again the sound is used in an instructionally relevant way to direct the student's attention to the critical characteristic of stress.

The computer provides a considerable arsenal of display possibilities. If these devices are used to focus the student's attention on critical aspects of the subject matter being presented then they can substantially facilitate the learning which occurs. If, however, they are used indiscriminately merely to be clever without a clear instructional intent they may distract from, rather than enhance, the learning which may occur. In short, **use attention focusing devices to relate examples to generalities and to point out critical characteristics of the illustrative material.**

Active Mental Processing

In an attempt to promote interactive instruction as opposed to linear presentations the early advocates of programmed instruction recommended that students should be required to respond overtly to the material being presented. Attempts to demonstrate the effectiveness of overt responding often were disappointing failing to show a distinct advantage for programs which required overt responses versus those which did not. More recent research has shown that it is not just overt responding that is important but the degree to which the student is required to mentally process the information which is being presented. Effective instructional conversations must engage the student in a way that requires thinking about the important aspects of what is being presented rather than passively observing information on the screen.

One of the techniques which promotes active mental processing is the use of rhetorical questions. A rhetorical question is one which is used to emphasize a point or introduce a topic and for which no answer is really expected. The advantage of using rhetorical questions rather than statements is that the student is more likely to engage his or her mental processes in formulating an answer rather than passively processing a statement. Then when this answer is confirmed by the presentation the point is more likely to be remembered.

Compare Figure 2, frame B, with Figure 6, frame A. The NOT-SO-GOOD program makes a simple declarative statement about the role of rhythm in distinguishing prose from poetry. The other program makes this same point but does it with a series of rhetorical questions ending with a rhetorical question which requires the student to actively engage in the conversation. The program does not wait for an answer to the first two questions: "What makes a poem a poem?" "Why is a poem different from prose?" The purpose of these questions is to cause the students to think about what they already know thus activating their mental processes so that when the point is made that one of the primary differences is the rhythm pattern they are anticipating this information and thus more likely to remember the point.

A related technique for promoting active mental processing is to engage the student in a conversation about the topic. Many educational programs use multiple choice questions as a way of interacting with the student. From a programming standpoint such questions are much easier to evaluate. However, in a conversation we are not interested in evaluating the student's answer anyway. The questions we ask or the response we prompt are another form of rhetorical question. We request an answer but the purpose is not to see if the student knows what we have said or is able to give the correct answer but rather the purpose is to cause the student to think. If the comment made by the student is something we anticipated we want to acknowledge that the student is thinking consistent with our intent. If on the other hand the student provides some statement that we have not anticipated we carry on much like we do in live conversations when we do not understand, that is, we merely restate our point with the hope that the student will then see where we were going. We do not say, "That is wrong!" or "I do not agree." We merely continue with the conversation.

Again compare Figure 2, frame B, with Figure 6, frames A through D. In Figure 6 frame A we anticipate that the student will make a response to the third request and the program provides a cursor to indicate to the student that we anticipate such a response. The program is written such that the student can type anything as a response. If the student is following our conversation, and is serious about interacting with the program, it is very probable that the response will be rhyme, since that is the most widely known characteristic of poetry. The program anticipates this response, including a dozen different ways to spell the word (rime, rhyime, etc.). In response to this input from the student the program displays the information in frame B by acknowledging the students input and suggesting that there is yet another characteristic which we are thinking about. If the student anticipates our intent and responds with meter, beat or rhythm (including likely misspellings) the program responds with frame C. If the student should respond with meter or rhythm to the first question the program displays frame C instead of frame B. Finally the program displays frame D which now makes the point in a declarative statement. If the response to either the question on frame A or frame B is unanticipated the program responds with frame D. The result of this conversational programming is that the student has the feeling that the computer is really carrying on a conversation rather than merely presenting information. Furthermore, the student is being required to think about the idea being presented.

Other examples of this conversatonal approach can be seen in figure 7, frames A. In this frame after presenting an example of poetry and an example of prose the student is asked to observe the difference. Our conversation anticipates that the student will observe the regular pattern of stress or meter. The program anticipates a wide range of key words expressing this idea. If the student uses any o, these key words in his or her response the program acknowledges with a simple, "Yes, passage 1 has regular or steady rhythm." If the student does not anticipate a key word then the "yes" is dropped from our reply but the rest of the statement is presented. The program then asks the student about passage 2. The conversation continues in this same way.

Figure 7, frame B provides yet another example of conversational programming. The program asks the student, "What is a stressed syllable?" Many students reply, "One which is stressed." This response does not go beyond the question so the program asks, "Can you think of another word?" If the student anticipates loud, emphasized, higher or synonyms of these words the program acknowledges with, "GOOD!" and the statement indicated on frame B. It should be noted that all of the displays are dynamic and the pieces shown come up one at a time so the second part of this display showing the passage of poetry actually appears after the student has been asked to read the passage aloud. Then the passage is chanted with sound enhancement to allow the student to compare his or her reading of the passage with the program's version of stressing the syllables.

Educational software should be conceptualized as a conversation rather than a lecture or a book. The interaction with the student is far more than merely asking the student if he or she is ready tc continue. The interaction should not be limited to asking the students questions for the purpose of seeing if they know the "right" answer. Rather the purpose should be to stimulate thought and actively engage the student in an interchange which will lead the student to understand what we, as an author, are trying to teach. In short, **promote active mental processing by asking rhetorical questions and engaging the student in a conversation which requires constructed (as opposed to multiple choice) responses to which we do not provide right-wrong feedback but rather an anticipation of a reasonable reaction much like that which would be expected from a live conversation.**

Examples

All of us know the cliche that there is nothing that teaches like a good example. For some reason the cliche has not found its way into the rules for effective educational software. The most common approach for tutorial programs is to immediately begin to ask the student questions, not of the rhetorical type discussed above, but questions testing the content which is being taught. If the student is uncertain about the content this approach often causes anxiety because the student does not know how to answer the question and being required to respond with the "right" answer when a student knows that he or she does not know causes frustration. Most educational software would be improved if the ideas were illustrated for the students by means of expository examples which show the application of the ideas being presented. Then, after the student has had an opportunity to study these worked examples, practice in applying the ideas is less threatening and more likely to be productive in promoting acquisition of the neccessary skills.

The NOT-SO-GOOD poetry program moves immediately from a page turning presentation of ideas to a series of questions asking the student to remember these ideas and then to a series of questions asking the student to apply these definitions to new words without seeing any examples prior to this practice exercise. This is not atypical of many educational software packages. In the improved version of poetry Figure 8, frame A shows one form of example presentation. The student has the opportunity to examine as many words of each type of meter as he or she feels are necessary to understand each of the meter types. Pressing the space-bar moves the pointer from one box to another. Pressing the return key causes the example word or phrase to appear in the box together with sound enhancement showing the stress pattern.

Figure 9, frame A shows a similar display for practice. After the student has seen as many examples as the student feels are necessary he or she can request the practice display by pressing the escape key. In this practice display words are presented one by one. The student uses the space-bar to point to the correct meter type. This is a form of multiple choice question. Note that in practice the intent is to determine if the student has mastered the concepts being taught. This display differs significantly from the conversational displays that were used to teach the ideas in the first place. If the student answers wrong the display indicates that the student is wrong, points to the correct answer and chants the word or phrase.

A similar set of example and practice displays are used for whole verses. The student is shown a passage of poetry. The dynamic display then shows the student how to divide each word into syllables, which syllables are stressed, where the poetic feet are divided, how the poem is chanted, and, finally, the type of meter involved. After examining as many of these expository examples as the student feels are needed to understand each type of meter a series of practice verses are presented to the student. After the student has classified each verse the option is given to do a detailed analysis of the verse.

Too often educational software moves immediately from the presentation to the practice without presenting illustrative examples of the ideas being taught. Students are able to learn more efficiently when they have the opportunity to study examples of the ideas prior to being asked to practice applying the ideas to new examples. In short, **provide expository examples as well as practice.**

SCREEN DISPLAY

This section identifies screen display techniques which are believed to facilitate the student's ability to interact with the materials. Screen display refers to the way that information is exhibited on the monitor. In this section we have attempted to identify a few display characteristics which may enhance or inhibit learning.

Scrolling

One of the advances in computer technology was the ability of a computer display to scroll material automatically as the text being entered reached the bottom of the screen. For many applications such as programming or word processing this automatic scroll is a real advantage which eases the user's task in entering material into the computer. However, what is an advantage for some applications is a disadvantage for others. When a student is trying to read a display it is very distracting if the text jumps up to the next line or worse jumps up several lines. The student loses his or her place and must then reread the material to find where he or she was. This interruption causes a distraction which often interferes with learning.

Students often use the location on the page as a cue to what was read. When reviewing material we often remember where it was on the page and use this spatial information to help us remember the content. When a page of text scrolls the material appears at a different place of the screen each time it is seen thus eliminating this spacial cue which often assists learning.

Look at Figure 2 again. The displays shown here illustrate scrolling. When the return key is pressed the material on the screen jumps up line by line as the new material is being written at the bottom of the screen. The student must read the text as it jumps or wait until the whole message is written before reading the screen. Also when the scrolling stops what was just read now appears in whole or in part at the top of the screen and the new material appears at the bottom of the screen. The student may be confused about what has been read and what is new.

While scrolling is an advantage for word processing, programming and other applications it is usually a disadvantage for educational programs. There may be some situations where it has an instructional purpose but usually it is an unnecessary distraction and should be avoided. In short, **no scrolling for educational programs.**

Text Output

In a text book the page is already filled with text when the student opens the book. With a computer display the test is written on the display before the student's eyes. Unimaginative programs often model the book and place a whole screen full of text up at once so the monitor resembles a page of text. Merely placing the text on the screen ignores the dynamic character of the computer screen and the possibilities for using text output to enhance the instructional effectiveness of the presentation.

One of the dangers of the dynamic display of text stems from the fact that students read at vastly different rates. When the text is removed from the screen under program control there is a danger that the student has not yet finished reading the message. This is like reading a book with someone who reads faster and having them turn the page when you are still in the middle. When the text is removed before the student has finished reading, the resulting confusion and frustration considerably inhibits learning. It is a good idea to allow the student to indicate when he or she is ready to continue (a good aspect of the "please press return to continue" message). Or, when the text is removed by the program there should be some means for the student to retrieve that which was removed.

Look at Figure 7, frame D. This is a dynamic display which presents the information a little at a time with a slight delay between each new part. The title comes up first, "FOUR KINDS OF POETIC FEET", followed by a 2 second delay. Then the four boxes appear. The headings are written one by one: "first syllable stressed", delay, "last syllable stressed", delay, "two syllables", delay, "three syllables", delay. Then the names of the four types of meter appear one by one: "TROCHAIC", delay, "IAMBIC", delay, etc. Then each type of meter is illustrated. The name is repeated, "TROCHAIC" with the definition, "2 syllables, 1st stressed". Then the direction, "LISTEN" flashes a few times along with the arrow pointing to the lower box. Finally, an example word is placed in the box one syllable at a time, while the stress is indicated by sound. After a delay of a few seconds the next type of meter is shown in the box until each of the four types have been illustrated. Examples of the different types of meter are presented and removed under program control. What if the student has not finished studying one type before the next appears? After all four types of meter have been illustrated a black stripe appears at the bottom of the display like the one shown in frame B. The directions, "[RET] REPEATS", means that pressing the return key will repeat the entire dynamic display again. In this way the student can control the text output even though individual elements are under program control.

For very short messages it is not necessary to provide means for the student to repeat the text presentation. For example, look at Figure 9, frame B. When the student chooses the wrong type of meter the program displays the message, "WRONG" and shows an arrow pointing to the correct choice. Then the program chants the correct version of the word. The program then goes to the next word. The feedback information is presented under program control. In this case allowing the student to tell when to remove the "WRONG" and go on requires an unnecessary response on the part of the student since there is no difficulty in understanding the feedback message.

Computer displays, unlike text, can use the dynamic presentation of text and graphic material to focus the student's attention and facilitate learning. However, if critical text is removed before the student has had a chance to finish studying the information, the result is frustration and interference with learning. A means should be provided for the student to indicate when he or she is ready to continue or for repeating information which is presented under program control. In short, **the student should control text output. Never erase critical information until the student indicates readiness to proceed OR provide a way for the student to repeat dynamically presented information.**

Dynamic Displays

Unlike a page of a text book, a computer screen is a dynamic medium. We have the added dimension of time which allows text to be displayed or erased at will. Furthermore, text can be black-on-white or white-on-black (inverse). Text can flash. Text or graphics can be animated and be moved about the screen to show relationships. Too many educational programs that assume these features are only for arcade games but do not take advantage of them for tutorial programs. These dynamic characteristics, when used in an instructionally relevant way, can do much to increase the effectiveness of the instruction.

Timing of text output can be used for stress or to improve readability. Look again at Figure 7. In frames A and B the stress patterns in poetry and prose are indicated by displaying a passage, syllable by syllable, where each syllable is accompanied by a high sound when stressed or a low sound when unstressed. Frame D used timing to emphasize the attributes of the various kinds of meter. Rather than putting the entire diagram up at once the attributes are put up one by one with a pause between so that the student will focus his or her attention on the attributes one by one. The names of the kinds of meter are displayed one by one so that the student will focus attention on each name as it appears. This timing of text output makes the presentation more easily understood because the student's reading pattern is directed rather than left to chance.

Flashing text can also be used to direct attention and prepare the student for that which is to come. Look at Figure 7, frames C and D. In these frames examples of the different types of meter are presented to the student one after another. If the words appeared without the direction to listen the student would not be prepared and would likely miss the use of sound to show the stress pattern. In this display the word, "LISTEN", and the arrow pointing to the box where the words will appear, flash for a second prior to the word appearing. In this way the student is anticipating the sound and the word when it appears.

If there is not an instructional purpose the use of these dynamic characteristics can be very distracting. Text which flashes or animation which which occurs with no apparent purpose distracts rather than enhances learning. Look at Figure 1. The cute cartoon characters which are used to indicate a correct answer soon lose their value. On the other hand the animation which occurs in Figure 4 enhances the idea which is being taught by directing the student's attention to the critical aspects of the display.

Computer displays, unlike textbooks, can be dynamic. Timing can be used to direct reading behavior, direct attention to particular portions of the screen or to stress certain ideas. In short,Xuse dynamic displays in which timing of text output, inverse text, flashing and animation are used for stress and emphasis.

Display Planning

When text books are put together a graphic designer often lays out the pages for aesthetic appearance and consistency. Effective layout can greatly facilitate the learning which can occur from the printed page. Too often, with computer displays, the need for graphic design is overlooked because the production process is simplified and steps which are usual in the production of books are eliminated. As a result too many programs look like rough drafts of printed material rather than carefully planned displays.

The screen should be planned as a unit. Each element, whether letter, word, or graphic, should be carefully placed on the screen at the location that the designer selects. Furthermore, the designer should have a purpose for placing information on the screen. If this purpose is aesthetic more pleasing appearing displays will result. If this purpose is instructional as well as aesthetic more learning will result. Following are a few characteristics of good screen design which should be considered. There is space here to identify only a few of the elements of good screen design.

Dark on light. The nature of a cathode ray tube makes turning on dots the natural way to write to the screen. Hence, white (or lighted) letters on a dark screen became the norm. After one has worked at a cathode ray tube for many hours lighted letters and a dark screen began to appear as normal. However, for a student who may be learning from a computer for the first time light letters on a dark screen seems very unnatural. They are accustomed to seeing dark letters on a white page. Consequently, it may be more desirable to paint the screen light and cause the print to appear as dark letters on a light screen.

Look at Figures 1 and 2 which use light letters on a dark screen. Does this seem confined or closed in? Now, look at Figure 6 which uses dark letters on a light screen. This gives an open less restrained feeling. Which appears more natural? It should be noted that for clarity on the printed page some of the NOT-SO-GOOD poet program (Figures 2 and 3) were printed in inverse. That is, the program used light letters and figures on a black backround. We feel these programs would be easier to read if the programs used the black-on-white appearance of the printed page as they appear here.

In short, **dark letters on a light screen will appear less confined and more natural to the student.** [3]

[3] It is interesting to note that the new APPLE Macintosh uses dark letters on a light screen.

<u>White space.</u> Printed media has cost. A book of 100 pages costs more for materials than a book of 50 pages. To conserve costs (not to mention ease of use) printed materials try to put more on the printed page. In fact, because of some unscrupulous home study businesses, the postal service has standards about how much print must appear on a single page of instructional material sent through the mail. It is unlawful to send materials through the mail that do not meet this minimum. You cannot make the materials look to be more by putting fewer words on a page. A computer display on the other hand is free, that is, 20 displays cost no more than 1 display in terms of paper. Hence, there is no good reason to fill the screen with text. Conversely, there are often good instructional reasons to display only what the student needs at a given time on the screen. After a given piece of text or graphic has served its purpose it should be removed. Leaving it on the screen merely clutters the display with unnecessary and possibly distracting information.

Look at Figure 2. Each of the screens are busy and full of text. Rather than inviting the student to read them they seem to promote the feeling, "Do I really have to read all that text?" Furthermore, after the student reads the first paragraph it (or worse, part of it) remains on the screen. This is unnecessary information which merely fills the screen and makes it more difficult to find the new paragraph.

Contrast the displays in Figure 2 with the displays in Figure 6. In Figure 6 the displays were carefully planned and sentences appear only where the designer intended for them to appear. There is plenty of white space between sentences making them easier to read. In contrast, the displays in Figure 2 were determined by the built-in computer functions of word-wrap at the end of the lines and scrolling. The author merely entered the information and let the computer determine the display.

In short, **leave plenty of white space and erase information when it is no longer needed.**

Short lines and natural phrases. In the early days, before 80 columns were common place, I have often heard designers of educational software complain that there was not enough space on the screen. Only 24 lines of 40 characters seems very confining when we are used to up to 60 lines of 70 to 80 characters. However, the discipline of stating things more concisely may actually improve the instruction if each of the ideas to be presented are carefully considered and the words carefully chosen. When, as a designer, you realize that it is not necessary to start every line at the left margin and fill to the right margin then you begin to realize that the placement of text on the screen can actually enhance comprehension. Since we are no longer constrained to use the page economically we are free to use the layout to convey the message that we are trying to present.

Look at Figure 7, frame B. Notice the response supplied by the computer to the rhetorical question, "What is a stressed syllable?" The stem of the statement is on the second line, "A stressed syllable is". Then each of the important ideas are expressed one per line, "louder", "emphasized", "higher in pitch". This listing tends to emphasize the individual characteristics so that they are not lost in a long string of prose. A similar use of short lines with a single natural phrase on each line is illustrated in the last part of this same frame. The stem of the statement is on the first line, "Stressed syllables are". The two ideas related to stressed syllables are emphasized by placing each on a separate line and indented from the stem, i.e., "higher in pitch" and "often found by reading aloud". A similar use of short lines with one natural phrase or idea per line is illustrated in frame C of this same figure. The stem appears on the first line, "A POETIC FOOT is". The characteristics are separated with one idea to a line and indented considerably, "a group of 2 or 3 syllables", "where 1 syllable is stressed", "and 1 or 2 are not stressed."

The Poetic Meter program sometimes does not use complete sentences when the idea can be presented with a list of abbreviated phrases. Some educators feel that educational software should demonstrate "good language use" and should therefore avoid sentence fragments and other incorrect usage. The use of short lines where one phrase appears per line can still be used as illustrated by frame C.

In short, **use short lines and separate natural phrases or ideas on each line.**

Fill justify. The thing that impressed me most when I first started to use a word processor was the fact that by merely giving a simple instruction I could have my document printed with right hand justification as well as left hand justification. The resulting document looked so professional, almost like it had been printed by a press. It has since come to my attention that some research has shown that a ragged right edge is easier to read. In fact the aesthetics of straight margins on both sides of the page may be detrimental to learning. What looks nice when a page is opened may in fact make it harder for the message to come across. If this is true for the printed page it is even more true for a computer screen.

In short, **don't fill justify on the screen.**

Upper case. In the early days of personal computers machines appeared which had no lower case alphabet. Many programs developed for these early machines were done in all upper case letters. Unfortunately, the BASIC programming language that still exists on many machines will allow lower case letters only with some difficulty. As a result it is sometimes easier to program in all upper case. However, the data shows that children often use the shape of the word as a cue to its meaning. When words appear as all upper case this cue is gone and reading becomes more difficult. There is no real excuse for professional programs to use all upper case and there is considerable evidence that all upper case presentations are more difficult to read and comprehend.

An isolated word or title in all upper case tends to make it stand out and is one form of creating emphasis. The use of all capitol letters for emphasis probably enhances the instructional impact and is desirable. But if all the letters are capitols this emphasis is lost.

In short, **do not present information in all upper case except for emphasis.**

Text style variety. The printed page is often made more pleasing by the inclusion of different styles of text. If this use of text style is haphazard and not related to the instructional purposes of the presentation it is probably distracting. On the other hand, use of a different style of text can enhance the presentation if it is used for emphasis or to indicate that the text appearing in the new style has a different purpose.[4]

The examples included with this paper have not made wide use of different text styles except the use of inverse (technically reverse inverse). In Figure 7, frame A the passages are shown as white-on-black to set them apart from the instructional text which is black-on-white. This is consistent throughout the program. All of the sample words and poetry are shown inside a black box as white-on-black. This helps the student to find the illustrative material and separate it from the instructional text.

In short, **use a variety of text styles to indicate different kinds of messages.**

HUMAN FACTORS

Human factors refer to those characteristics of the software that make it easy for the student to use. The difficulty of using many

---

[4]Newer personal computers allow different text styles on the screen even for ordinary document creation.

computer programs has become recent folklore. It is frequently necessary to memorize a long list of control characters or other commands in order to interact with the program. The most recent advances have attempted to deliver software that requires little or no knowledge of computers; software with which the user can interact by pointing. In this section we have suggested some of the human factor concerns which will make educational software easier to use and thus result in more efficient learning.

Control of Location

Many educational programs have the characteristics of an adventure game. The student has wandered into a maze and cannot see the way in or the way out. The only choice available is to keep working in the hope that the light at the end of the tunnel will soon appear. The problem with this approach is that the student may find that he or she already knows what is being presented and would like to skip ahead to the next section. Or, worse, the student is having difficulty and would like to return to a previous section. Many students have learned that a quick survey through the material before beginning more serious study greatly facilitates the ability to integrate the new information with that which is already known. When there are no indicators as to where you are or where you are going this survey activity is impossible.

By the use of indexes and tables of contents, books enable us to preview, survey, find an isolated piece of information, skip ahead or review. If we do not provide similar devices in educational software we have created not more flexible but less flexible educational materials. Some possible devices for allowing more student control over the materials are as follows:

Table of Contents. Like a book, a table of contents is a useful tool for an educational program. This should list the major topics to be covered and provide a way for the student to "turn to" a given topic. Look at Figure 5, frame B. In the Poetic Meter program after the title page the student is presented with a list of the sections of the program. In this program the student is required to keep track of his or her own progress (there is no student manager program) and this table of contents was necessary to enable the student to pick up where he or she left off. But even if this was not necessary, having the table of contents as an option would facilitate preview and review.

In short, **provide a way for the student to skip to the major sections of the program in order to preview, review, or repeat portions of the material.**

Location indicators. A table of contents provides a map of where to go and the mechanism for getting there, but do you know where you are when you get there? The size of a software program is transparent to the student. It is hard to turn to the last page and see how much more you have to study before it is convenient to take a break. Books use page numbers. On the table of contents page you can observe that a given section is 10 pages long. Then by monitoring the page numbers you can tell how much of the section is left. Some similar concept is needed for educational software. Perhaps a page number is sufficient.

Look at Figures 6 through 11. Notice that each display has a number at the bottom of the screen. This is an attempt to tell the student where he or she is in the program. By comparing this number with the table of contents the student can determine where he or she is in the total program. In short, **provide some sort of location indicator so that the student knows where he or she is in the total program.**

Pages. When studying a book if a student gets distracted it is easy to turn back to the last page and reread the material to pick up on the train of thought. Too often in educational software after the return key has been pressed the last presented material is gone, never to be retrieved unless the program is started over. While a table of contents allows preview or review of whole sections of material, there is need for a mechanism for short term review or preview. The student must be able to "turn back" to the previous page, to "turn forward" to the next page, or in the case of dynamic displays, repeat the current page.

Look at Figure 6, frame D and Figure 7, frame B. At the conclusion of each page in the Poetic Meter program the student is given the option of going back to the last page by pressing the left arrow key, going on to the next page by pressing the right arrow key or repeating the current page by pressing the return key. In short, **allow the student to "turn" the pages by going back to the last page, repeating the current page or going forward to the next page.**

Keyboard Anxiety

More people are afraid of keyboards than are afraid of computers. Typing has been seen as a skill for typists (a job description rather than merely a qualification), clerks and secretaries. It is still taught as an elective course in high school (usually in the business department). It is apparent that like handwriting typing will be a basic skill in the future but for the time being we must realize that the majority of the population do not have keyboard skills. This means that lengthy keyboard input is likely to cause anxiety and frustration. Therefore programs should be designed that minimize the amount of information required from the keyboard.

The latest technology in computer programs uses pointing as the primary input device. Pointing, using a mouse[5] or an easily located key, is much easier than finding a particular letter or number on the keyboard. Pointing is much easier than typing a word or phrase. There are times when we want unstructured input from the student. Figure 6 represents this type of situation. Other times when we are involved in selecting a menu or choosing a response to a question pointing will facilitate the student's interaction. Figure 5 illustrate pointing on a menu. The student merely presses the arrow keys to move the pointer up and down the menu. When the arrow points at the choice that the student wishes to select he or she merely presses the return key. A similar pointing procedure is used in selecting examples and answering practice

[5]A mouse is a hand held device which rolls on a desk moving a pointer on the screen. A button on the mouse selects the function which the arrow points to on the screen.

problems. Figure 8, frame A, shows the use of a pointing device for selecting the type of meter the student wishes to see illustrated. Pressing the space bar moves the pointer (the brackets around the choice in this case) which rotates past each of the choices in turn. The return key gets the choice. Figure 9, frame A shows a similar pointer for practice. In this case the student uses the space bar to select the type of meter illustrated. Pressing the return key accepts the student answer.

Many students get frustrated when required to use the keyboard and this anxiety interferes with learning. In short, **minimize unnecessary typing by using a pointing device whenever possible.**

Advisor

One aspect of effective teaching is to monitor the student's process and progress and to provide advice for the student to guide further study efforts. This type of monitoring function is difficult to implement in the classroom where there are usually many children for every teacher. One advantage of the computer is that it can not only present information to the student but it can gather information from the student. This student generated information can then be used to monitor the student's progress and can serve to trigger advice to the student which may facilitate his or her learning.

The most obvious thing to monitor is the number of practice problems that a student misses. However, there is much other information which can be used to determine if the student is using the instructional program in an effective way. Some of these include monitoring the number of examples studied by the student, checking to see if the student has used the resources provided by the program as well as checking on practice performance.

Look at Figure 8, frame B. This display is shown to the student if he or she presses the escape key to terminate the example portion of the program before we as the designer feel that an adequate number of examples have been studied. In fact there is a prior message in this section of the program. If the student tries to escape from the example section before looking at a minimum of two examples of each type a message says, "You must study at least 2 examples of each type." In this example display the amount of prompting material is slowly decreased. Early examples are shown divided into syllables and the word or phrase is chanted for the student. After a few examples the colon which divides the words into syllables is removed and the words are no longer divided into syllables for the student. After a few more examples the chant is turned off and the student merely sees a word or phrase of the type indicated without additional prompts. If the student escapes before seeing at least 2 nonprompted examples of each type the warning message of Figure 8, frame B appears. If the student does exceed this criteria then danger frame is replaced by a frame which say, "GO, you should be ready for practice." Notice that in either the case of the danger or go advice the student has the option to follow the advice or ignore the advice. The black stripe at the bottom of the page gives the student the option of returning for more examples by pressing the left arrow key or going on to practice by pressing the right arrow key. If the student elects to return to the example section of the program the software remembers where the student left off and begins presenting examples from this point.

A similar advisor function follows practice. Look at Figure 9, frame C. In this case the student is given a score board showing how many of each type of meter were classified incorrectly. If the student misses more than 2 of any type the warning includes a stop sign and the advice, "You should study more examples." If the student misses less than 1 of each type the display contains a "GO" sign and the advice, "OK to study the next page". If the performance is between these two criteria levels the display contains a "CAUTION" sign and the advice, "Perhaps you should study more examples." The student is given three options. Turn to the next page by pressing the right arrow. Go back to the example section by pressing the left arrow. Do some more practice by pressing the return key. Again the student has the option to override the advice.

Computers provide the mechanism to monitor not only the student's performance but all of the student's learning activities. The teaching function of advice can easily be implemented. In short, **monitor the student's activity and provide advice when potentially decremental action is taken. In most cases provide a mechanism for the student to override the advice.**

Learner Control

In the early days of computer assisted instruction advocates often stated that someday we would have a computer system which could assess the readiness and aptitude of the student and then present that information which was an optimal match to the individual student. Even if it were possible there remain some serious ramifications of this assumption. If a significant part of the curriculum was constructed around such adaptive instruction the gains on the part of the student in the subject matter being taught might be considerable but the side effects could be catastrophic. The students in such a system would be spoon fed and might never learn to feed thenselves. The real world is seldom so accommodating. If one of the primary aims of education is to assist the student to be an independent learner and acquire additional learning by self teaching then our highly adaptive CAI system may turn out to be maladaptive after all. Students who have a major part of their learning on such an accommodating system would not learn the important metaskills necessary to learn from less than optimal information.

A better approach might be to teach the student to adapt the system to themselves rather than for the system to adapt itself to the student. The mechanism for this student adaptation is to provide learner control so that many decisions about the next best display are left to the student. Rather than big brother providing the information thought to be best for the student the student must select that information that he or she thinks is best for himself or herself.

Text output. In an earlier section we discussed student control of text output. This is an obvious area where students can determine for themselves when they are ready for the next display. Exceptions might be in timed tests or when speed reading is the objective but usually the student should indicate, "I'm ready to continue."

Number of examples. Research has shown that most students can determine how many examples they need to study in order to understand an idea. The number of examples required by different individuals may differ considerably but if an individual is shown less than he or she feels is needed there will be a decrement in performance. On the other hand, showing them more than they think is needed serves only to prolong the instruction without any accompanying increase in performance.

Look at Figure 8, frame A. In this presentation the student can determine the number of examples needed to understand each type of meter. When the student feels that enough has been studied he or she presses the escape key and jumps to the practice section. Note that the advisor function previously discussed comes into play to monitor the student's use of learner control. If in our opinion the student might be using learner control to avoid what might be necessary study the program provides a caution to the student. However, the student is still in control and can usually override the advice.

In short, **allow students to select how many examples they need to study.**

Optional help. Students differ significantly in the amount of elaboration they need to understand an idea. Some students need a step by step explanation while others understand after the general idea has been presented or after they have seen an example. It is inefficient and may cause frustration for the student if forced to work through every step of a detailed analysis. On the other hand if this detailed analysis is not available some students will be lost.

Look at Figure 10. This sequence of frames illustrates the detailed analysis of the poetry scanning procedure. First, the student is advised to read the poem aloud and take a guess the type of meter involved (frame A). The scanning editor then "chants" the poem for the student using the sound to indicate the stress pattern involved. In frame B the student is advised to separate multisyllable words into separate syllables using the colon. In frame C the student is advised to mark the stressed syllables using the underline. And in frame D the student is advised to divide the passage into poetic feet using the up arrow. Finally the student is asked to observe the pattern of stress in each foot and indicate the type of meter involved.

Some students need to repeat this detailed analysis many times while others will get the idea very quickly. Students can jump to the practice at any time by pressing the escape key. In the practice section (see Figure 11) the student is asked to classify the type of meter involved in the passage (frame A). In frame B the word **RIGHT!** or **WRONG!** is flashed in the upper right part of the screen to indicate whether or not the classification was correct. The passage is then chanted for the student to allow comparison. The correct choice is not indicated to the student when he or she is wrong. The student can then choose whether or not to engage in the more detailed analysis. Pressing the [?] key causes frame D to appear and then follows a detailed scanning analysis such as was presented in the example section.

Because students differ in the amount of detailed elaboration they need they should be given a choice as to whether or not they wish more exaplanation. Forcing all students through a linear path consisting of the same level of detail may be boring for some students because the pace is too slow and frustrating for others because the pace is too fast. In short, **provide optional help, don't force every student through the most detailed presentation.**

Escape. Sometimes students are interrupted while studying. There are many causes for such interruption.. If a student is in a long sequence he or she is very hesitant to merely pull out the disk and turn off the computer. This may mean that they must start over the next time or they may fear that this will somehow ruin the program. The program should provide a way for the student to return to the menu or some other starting place.

Poetic Meter allows the student to escape from any sequence by pressing the escape key. Usually the consequence of such an action is indicated by the program. In the example sections the program uses the escape as a means of learner control for the student to tell the program that they have seen enough examples and wish to go on. The advisor warns the student if we feel that the student should study more examples. This program does not have a student manager so if the student uses escape to leave the program the program starts over when the program is restarted. Including a student manager is better. This provides a way for the program to restart the student where he or she left off without the student having to keep track and use the menu to find his or her place.

Everyone suffers from interruptions and needs a way to exit in an orderly way short of pulling the plug. In short, **provide a means of escape from any lengthy activity but advise the student about the consequence of such an escape.**

Directions

Those of us who have been in the computer business for some years have come to accept a number of conventions that seem like second nature. For example, everyone that has used a computer very long soon learns that to enter a response it is usually necessary to press the return or enter key. This is so obvious to us that we usually feel that to inform the student about this convention is unnecessary. However, more and more first time users are beginning to use the computer. What may be obvious to an experienced user is not at all obvious to the first time user. The learner will have a more pleasant experience if the program contains complete directions about what to do and when.

Often documentation will contain the necessary directions but if they are at all complex, as in editors where there are several options available, most users find it is necessary to have a summary card in order to remember which commands go with the program. If possible, the current options should be available to the student as options on the screen.

Look at Figure 9, frame A. This is a simple exercise but the directions for the student are shown on the screen so that what is necessary to make the selection is clearly indicated. The Poetic Meter program makes a poor assumption on Figure 6, frames A and B. It assumes that the student will know that when the cursor appears he or she should enter an answer and then press the return key. It would be an improvement if this direction appeared on the screen at least for the first few times the student is expected to make some constructed response to a request from the program.

Don't create interfering directions. Sometimes in an attempt to make it very clear to the beginning user the procedures become very inconvenient for the experienced user. This is probably less of a problem in an educational program than it is in some utility such as a word processor or data base. Nevertheless we should be careful to balance directions so that the program is easy to use for the beginner but convenient to use for the more experienced user.

Make the directions as natural as possible. There is nothing more difficult than a long menu that is numbered. The user must look at the menu, get an arbitrary number, and hunt for the number on the keyboard in order to make a selection. Mnemonic choices using the first letter of the word may be easier to remember. Pointing, as already mentioned, is easiest of all. Keys should be used in a natural way or in ways consistent with other systems. Unique combinations of key presses make it unnecessarily difficult for the student. Even the advice to the student should use already known symbols. In the Poetic Meter program the advisor uses "STOP", "CAUTION" and "GO" as ways to get the student's attention quickly and to leave little ambiguity about what is being said. (See Figu _, frame B and Figure 9, frame C.)

In short, **provide adequate directions including all of the options available to the student. If possible make these available on the screen or accessible with the press of the [?] key. Use the most natural procedure.**

Program Structure

This paper has not dealt with programming design but it is necessary to border on the architecture of the program for one final human factors concern. While the power and capacity of personal computers seems to increase daily with each new product on the market, there are still severe limitations of space and size. These limitations can interfere with effective educational design. Some popular programs are very annoying because they continually access the disk drive. For the student it seems like half the time is spent waiting for the computer to load the next part of the program. The polite message, "Thank you for waiting." gets very old in a hurry. By carefully structuring the program these long delays in processing or accessing information can be minimized. It is especially important that there be no delays between the time when a student enters information and the computer reacts to that information. If there is a delay the student often feels that the computer is not operating properly and may try to enter additional information. Whenever possible the next part of the program should be anticipated and loaded during a time when the student is busy with other activities on the screen. ...s when the student indicates that he or she is ready to continue the next part of the program is already in memory and easily displayed to the student.

In short, **plan disk access to avoid long waits while the computer retrieves information.**

In Conclusion

Education Secretary Terrel Bell has recently criticized educational software as "electronic page turning [that] hasn't been designed to do a good job of interacting with the mind of the student." He further states that computers could be used as "slave mechanisms" to check spelling, punctuation and sentence structure but he doubts that computers could ever teach writing. As a designer of educational software (including software that teaches writing and poetic meter) I was ready to go to battle with the good Secretary. Then I realized that a random sampling of educational software might very well create the impression that computers are pretty limited in what they can teach. It is time that the consumers of educational programs become more discerning and refuse to purchase software that does not meet at least the minimum standards of acceptable instruction. This paper has attempted to provide a few suggestions about some of those minimum standards.

SUMMARY

Instructional Strategies

The first section of this paper identifies design strategies which are believed to have a direct impact on the amount of learning that occurs as a result of instruction. Instructional design strategy involves the arrangement of content elements and other information in such a way that learning is facilitated. The following guidelines are suggested:

1. Avoid merely putting text on the screen; avoid mere page turning.
2. Avoid generality-rich but example-poor presentations. Each idea should be presented by a generality, examples and practice.
3. Avoid remember only practice.
4. Use attention focusing devices to relate examples to generalities and to point out critical characteristics of the illustrative material.
5. Promote active mental processing by asking rhetorical questions and engaging the student in a conversation which requires constructed (as opposed to multiple choice) responses to which we do not provide right-wrong feedback but rather an anticipation of a reasonable reaction.
6. Provide expository examples as well as practice.

Display Techniques

The second section of this paper identifies screen display techniques which are believed to facilitate the student's ability to interact with the materials. Screen display refers to the way that information is exhibited on the monitor. The following guidelines are suggested:

1. No scrolling for educational programs.
2. The student should control text output. Never erase critical information until the student indicates readiness to proceed OR provide a way for the student to repeat dynamically presented information.
3. Use dynamic displays in which timing of text output, inverse text, flashing and animation are used for stress and emphasis.
4. Dark letters on a light screen will appear less confined and more natural to the student.
5. Leave plenty of white space and erase information when it is no longer needed.
6. Use short lines and separate natural phrases or ideas on each line.
7. Don't fill justify text on the screen.
8. Do not present information in all upper case except for emphasis.
9. Use a variety of text styles to indicate different kinds of messages.

Human Factors

The third section of this paper identifies some human factors characteristics which will make educational software easier to use and thus result in more efficient learning. The following guidelines are suggested:

1. Provide a way for the student to skip to the major sections of

the program in order to preview, review or repeat portions of the material.

2. Provide some sort of location indicator so that the student knows where he or she is in the total program.
3. Allow the student to "turn" the pages by going back to the last page, repeating the current page or going forward to the next page.
4. Minimize unnecessary typing by using a pointing device whenever possible.
5. Monitor the student's activity and provide advice when potentially decremental action is taken. In most cases provide a mechanism for the student to override the advice.
6. Allow students to select how many examples they need to study.
7. Provide optional help, don't force every student through the most detailed presentation.
8. Provide a means of escape from any lengthy activity but advise the student about the consequence of such an escape.
9. Provide adequate directions including all of the options available to the student. If possible list the available options on the screen or make them accessible with the press of the [?] key. Use the most natural procedure.
10. Plan disk access to avoid long waits while the computer retrieves information.

FIGURES

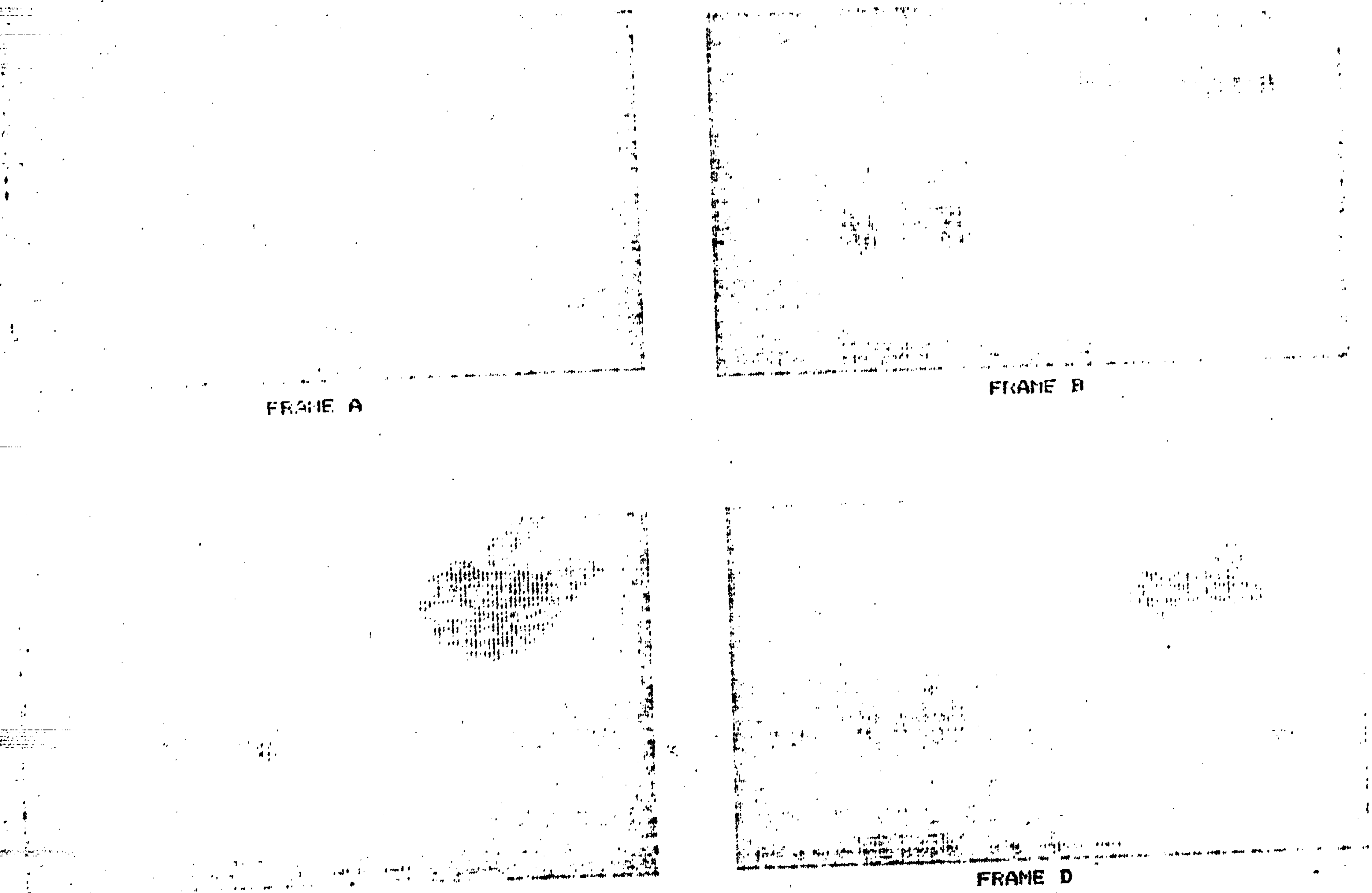


Figure 1 Drill and practice program.

FRAME A

FRAME B

The four kinds of stress patterns to be taught can be represented by the following table:

	FIRST SYLLABLE STRESSED	LAST SYLLABLE STRESSED
TWO SYLLABLES	TROCHAIC	IAMBIC
THREE SYLLABLES	DACTYLIC	ANAPESTIC

Return to continue.

FRAME D

FRAME C

Figure 2 NOT-SO-GOOD Poetic Meter

[illegible] FOOT consists of a stressed syllable followed by an unstressed syllable. A single foot is called a TROCHEE.

Return to continue

An IAMBIC FOOT consists of an unstressed syllable followed by a stressed syllable. A single foot is called an IAMB.

Return to continue.

FRAME A

For each of the [illegible] the correct [illegible] button after your [illegible].

1. An anapestic foot consists of

a. 1 stressed followed by 2 unstressed syllables.
b. 1 stressed followed by 1 unstressed syllables.
c. 1 unstressed followed by 1 stressed syllables.
d. 2 unstressed followed by 1 stressed syllable.

a
NO! Please try again.

FRAME B

d. 2 unstressed followed by 1 stressed syllable.

NO! Please try again.

d
THAT'S RIGHT !

2. An iambic foot consists of

a. 1 stressed followed by 2 unstressed syllables.
b. 1 stressed followed by 1 unstressed syllables.
c. 1 unstressed followed by 1 stressed syllables.
d. 2 unstressed followed by 1 stressed syllable.

a
NO! Please try again.

FRAME C

For each of the following words indicate the kind of meter by typing T for trochaic, I for iambic, D for dactylic, and A for anapestic.

jaunt.y i

WRONG 0 right

glut.ton.y d

CORRECT 1 right

sa.lute i

CORRECT 2 right

FRAME D

Figure 3 NOT-SO-GOOD Poetic Meter continued

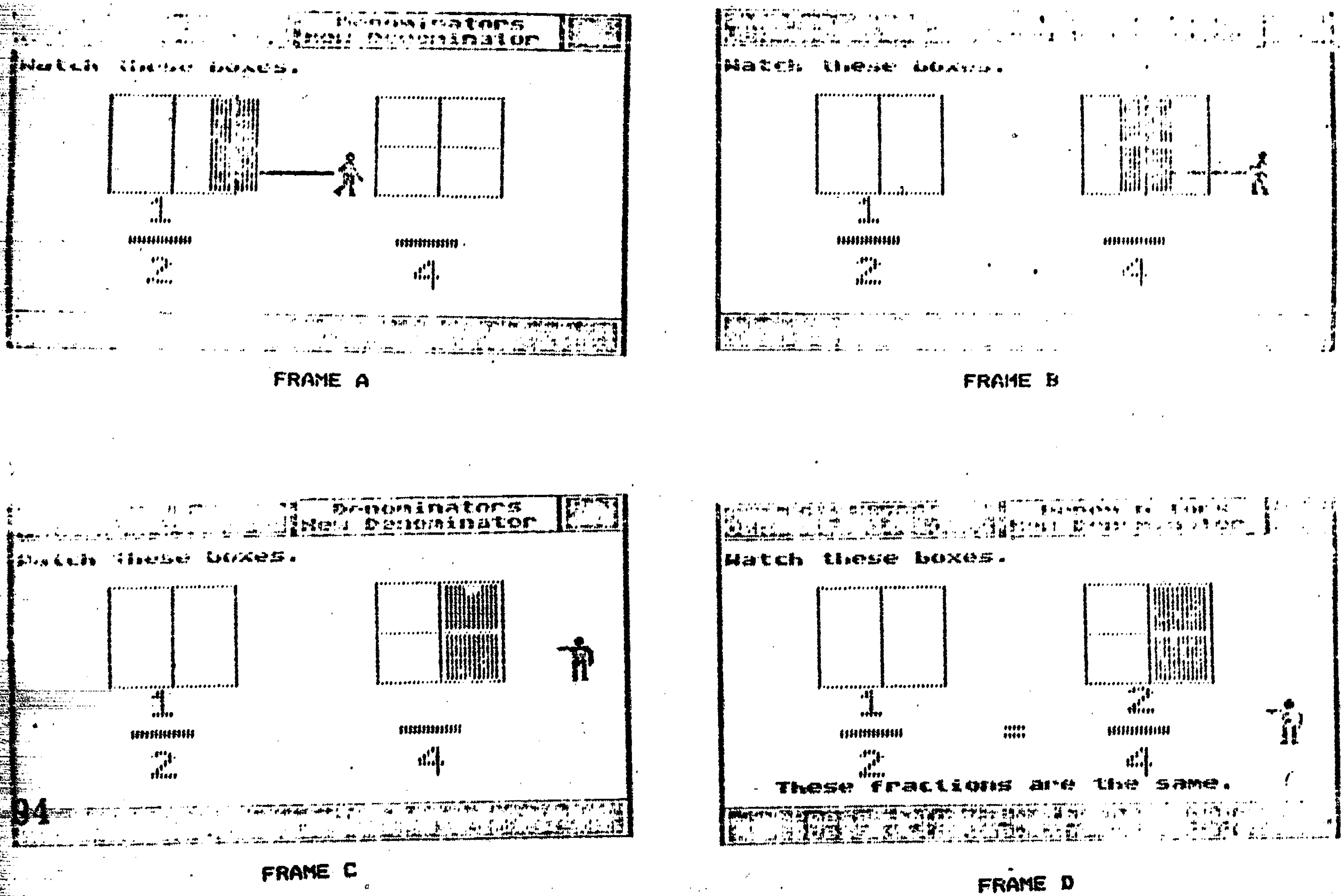
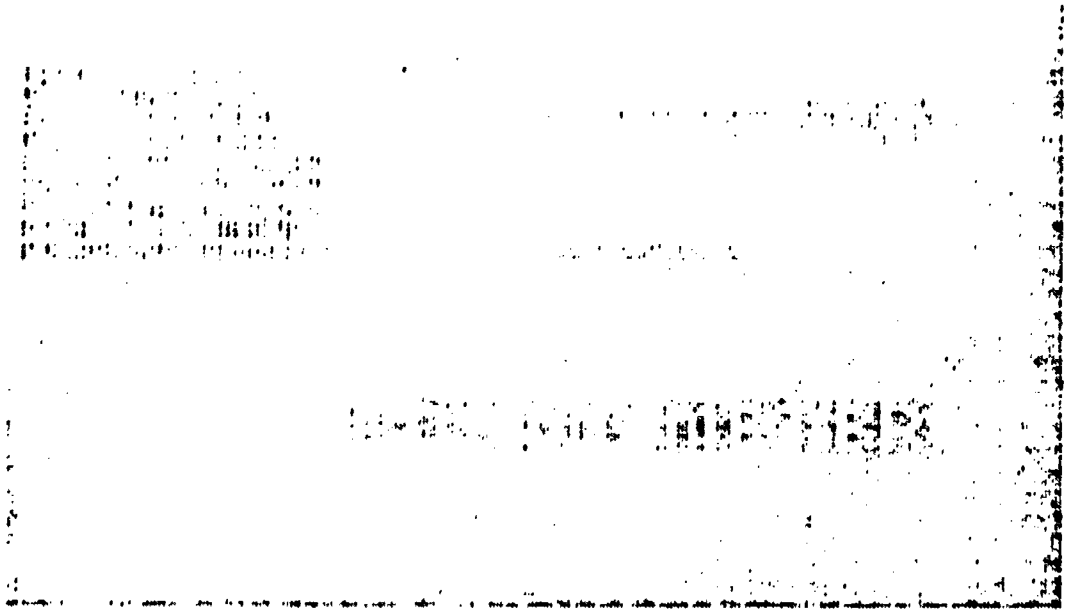
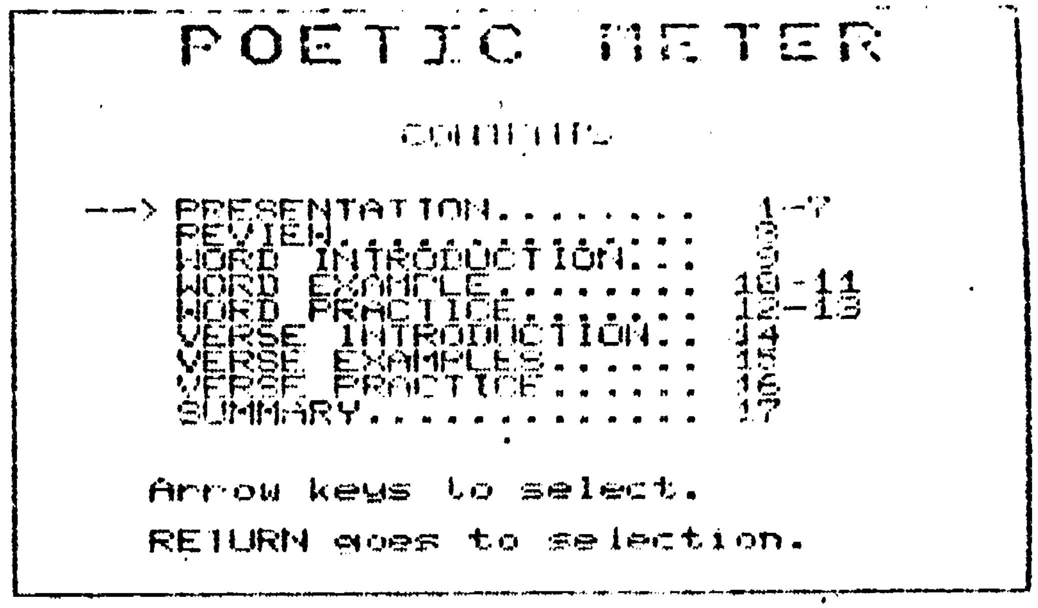


Figure 4 EDU-WARE FRACTIONS with animation



FRAME A



FRAME B

Figure 5 Table of Contents - Poetic Meter

POETIC METER

What makes a poem a poem?
Why is a poem different from prose?
Name one characteristic of a poem.

1

FRAME A

POETIC METER

What makes a poem a poem?
Why is a poem different from prose?
Name one characteristic of a poem.

RHYME is one characteristic
Can you name another?

1

FRAME B

POETIC METER

What makes a poem a poem?
Why is a poem different from prose?
Name one characteristic of a poem.

RHYTHM or METER is very important.

98

1

FRAME C

POETIC METER

What makes a poem a poem?
Why is a poem different from prose?
Name one characteristic of a poem.

POETRY has two main characteristics

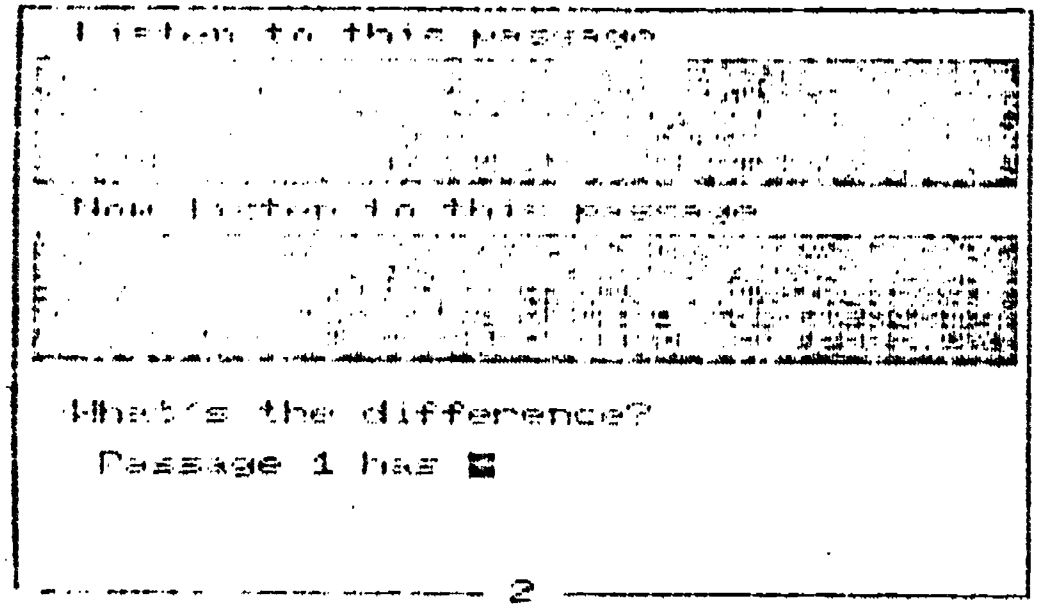
RHYME and RHYTHM

99

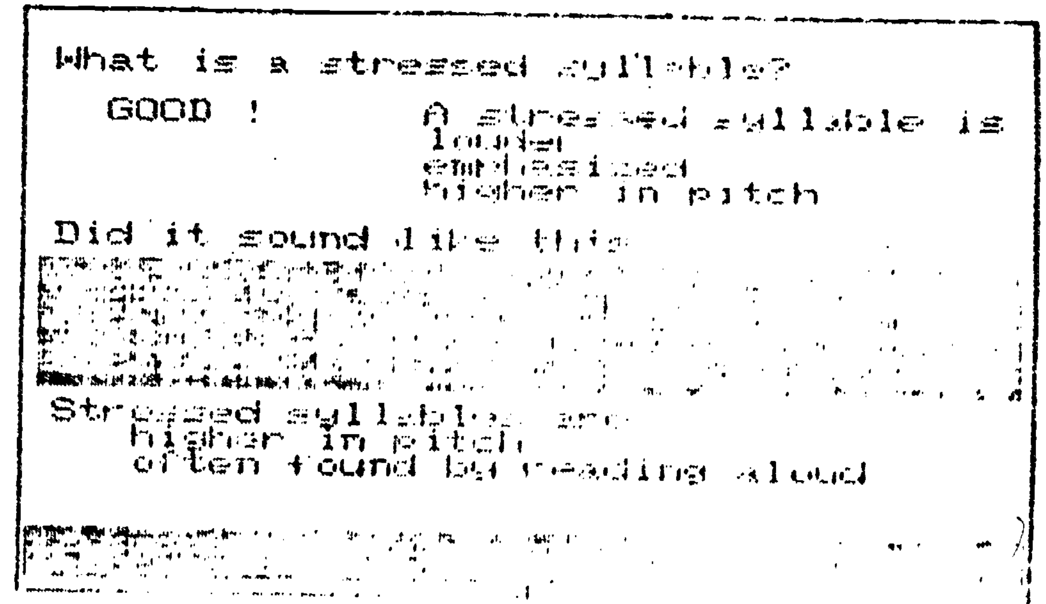
1

FRAME D

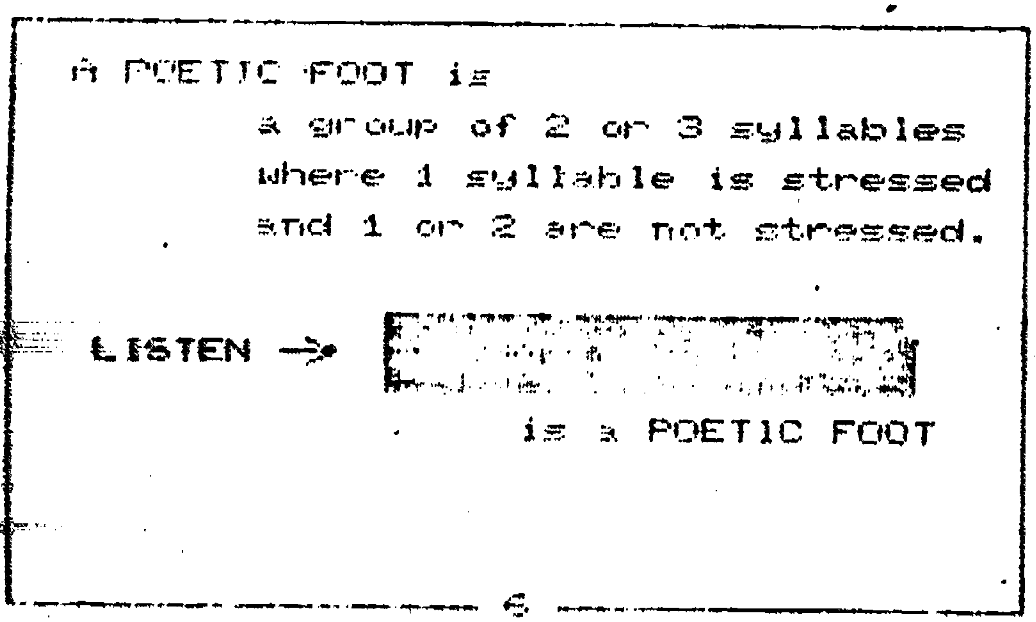
Figure 6 Presentation page 1 - Poetic Meter



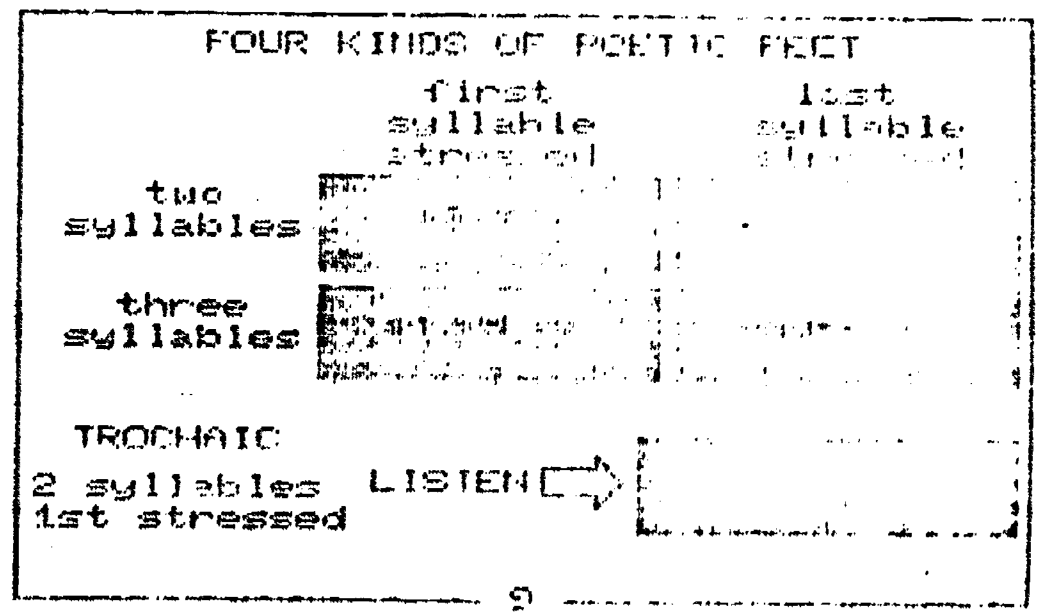
FRAME A



FRAME B

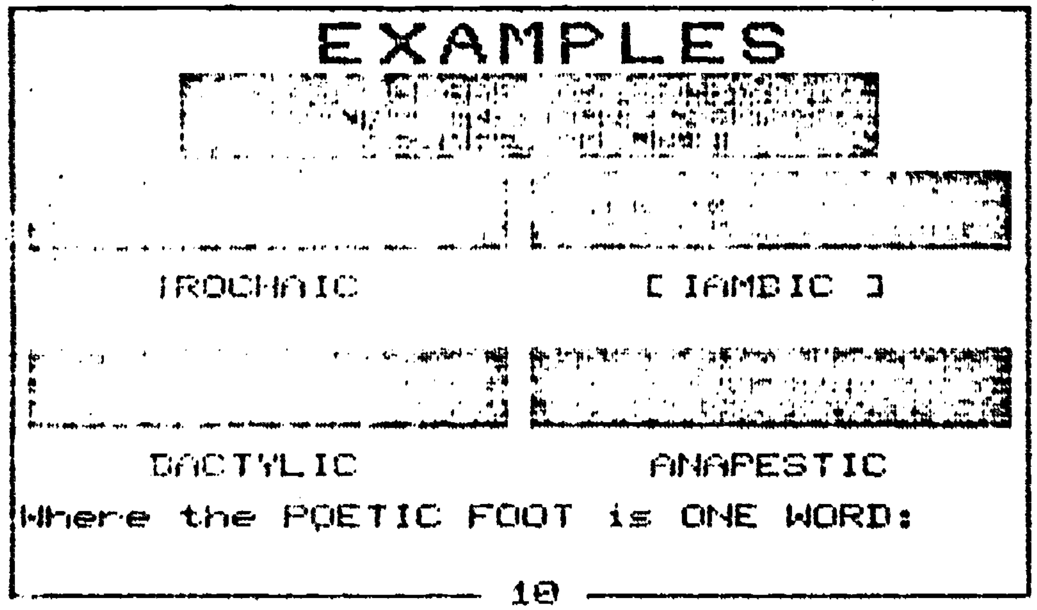


FRAME C

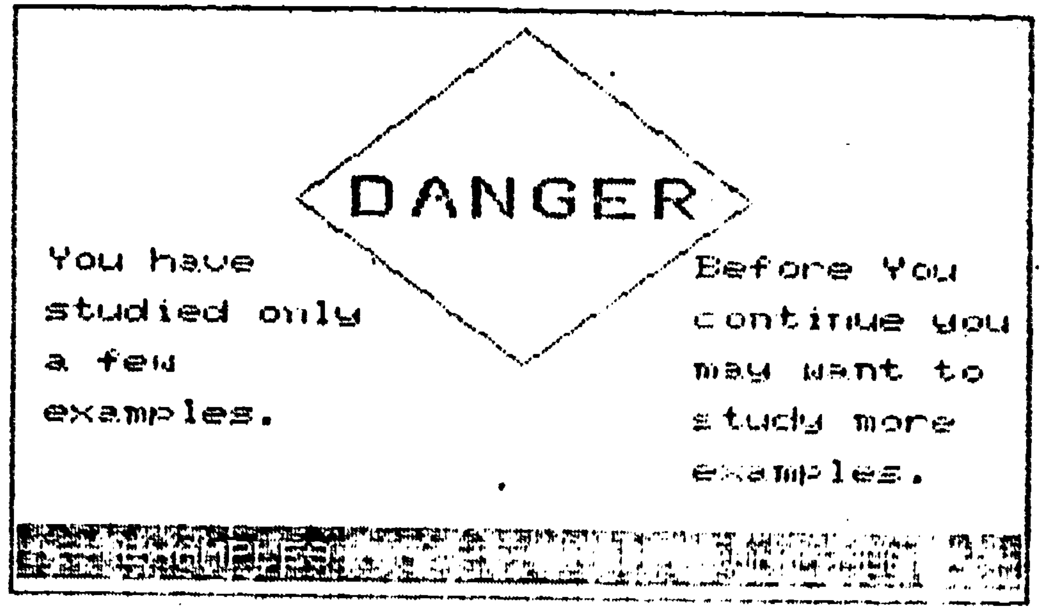


FRAME D

Figure 7 Presentation selected pages - Poetic Meter

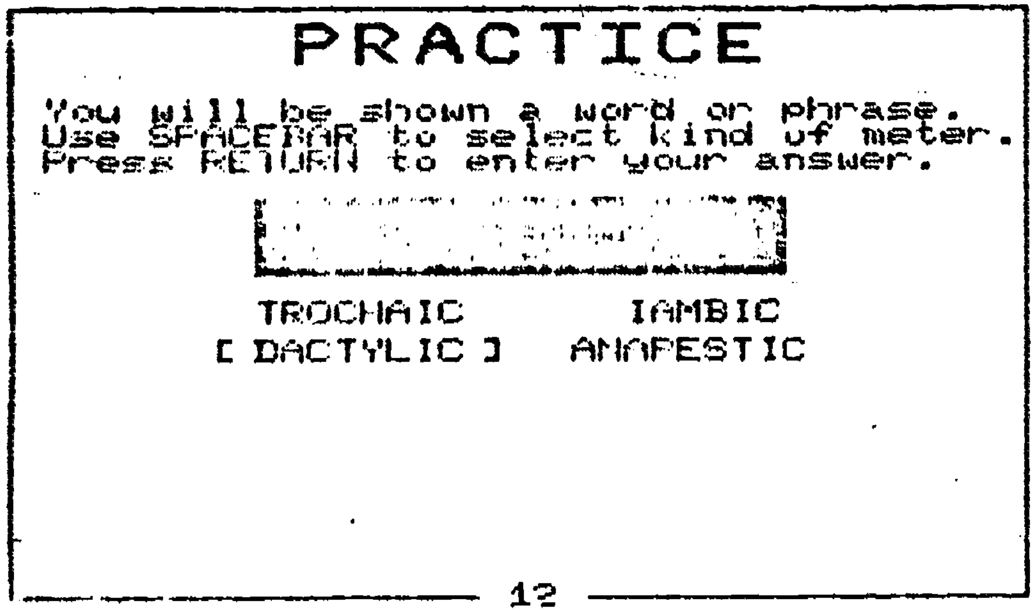


FRAME A

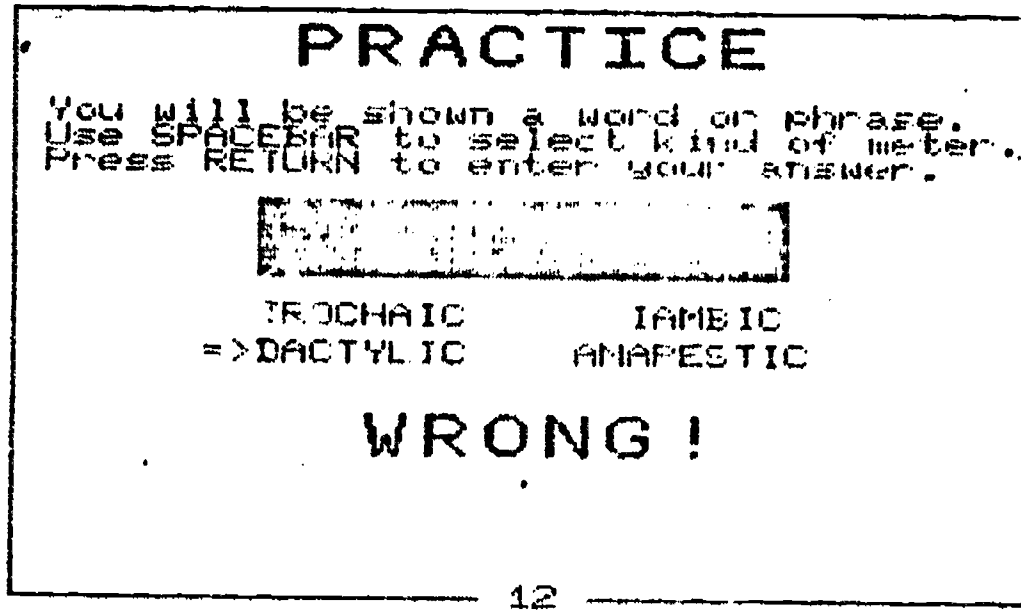


FRAME B

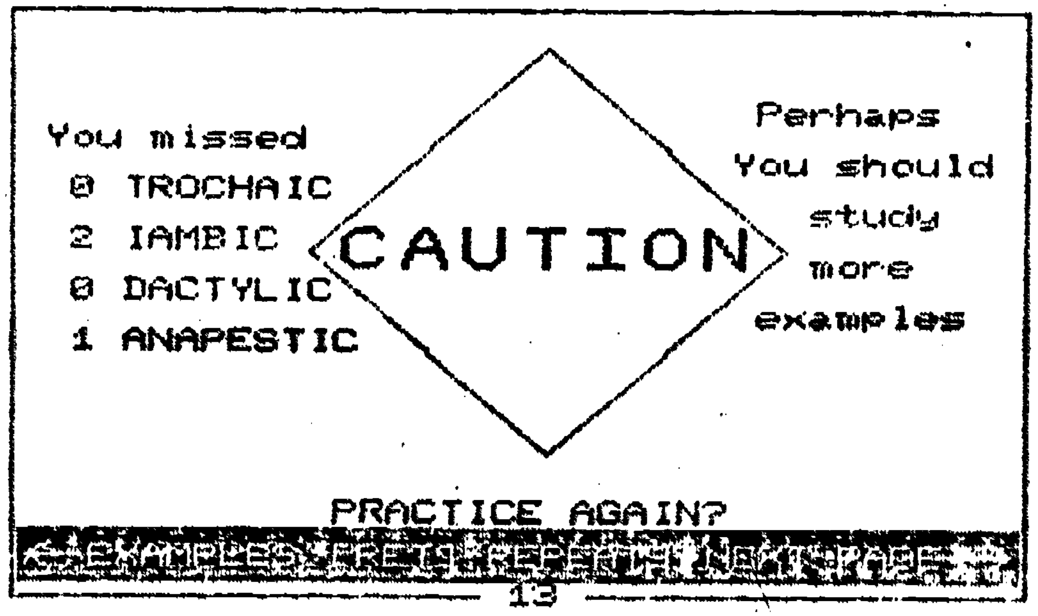
Figure 8 Word Example - Poetic Meter



FRAME A

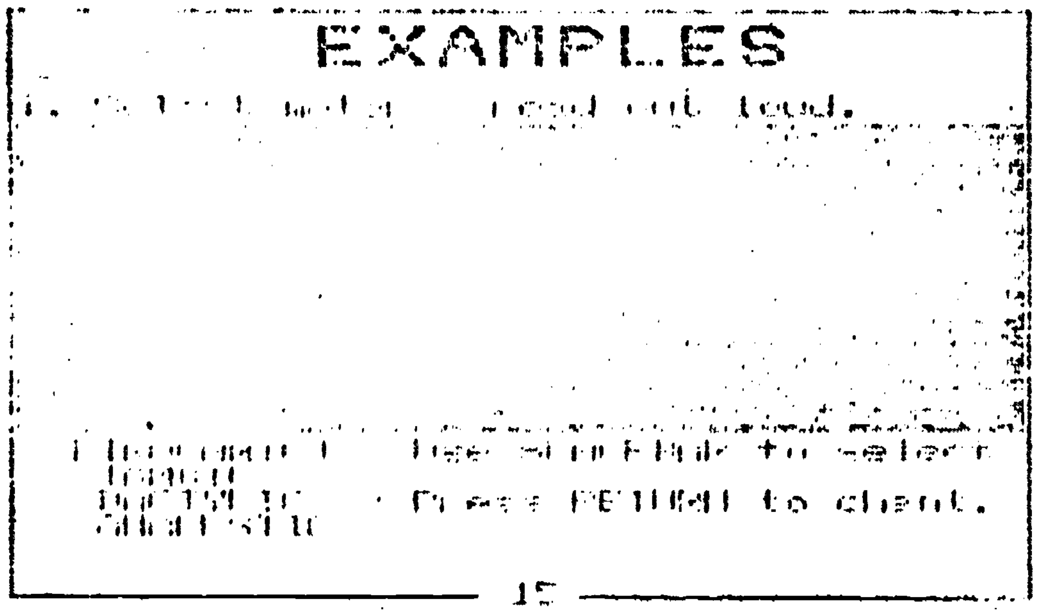


FRAME B

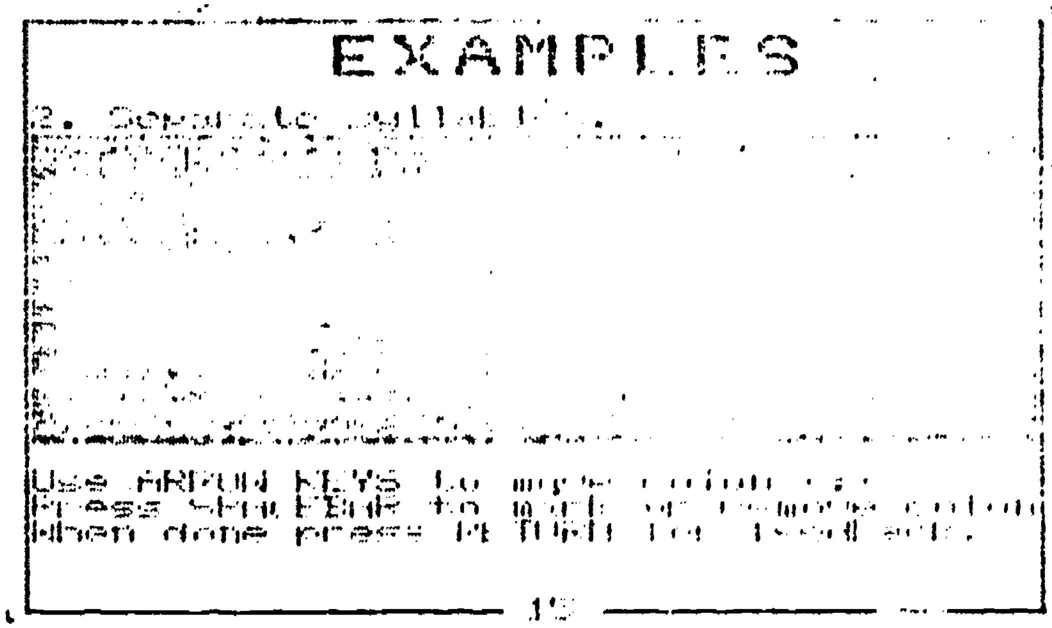


FRAME C

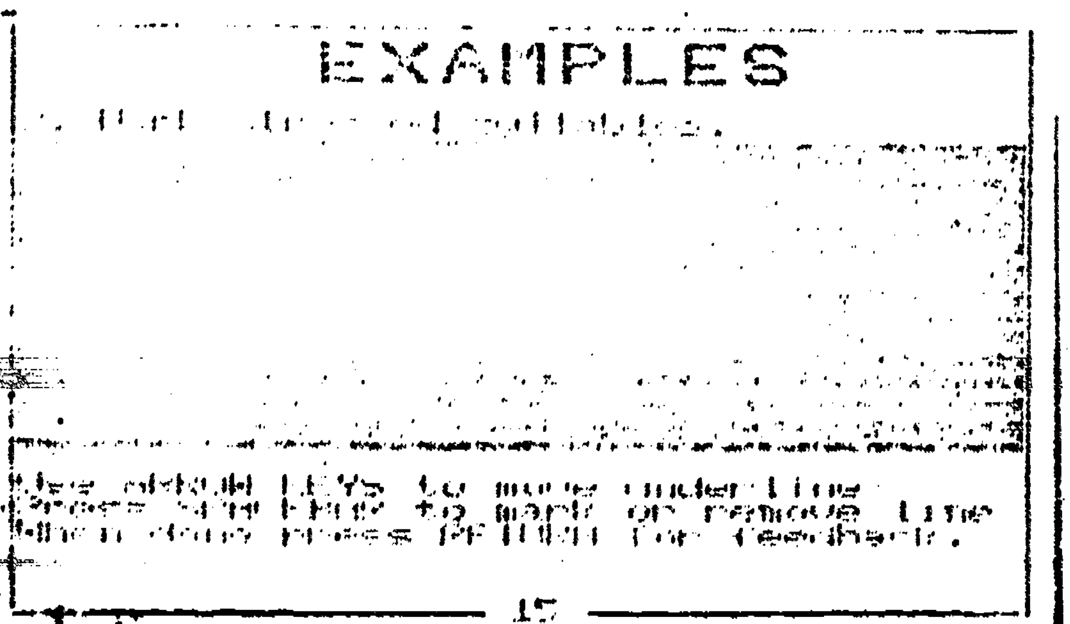
Figure 9 Word Practice - Poetic Meter



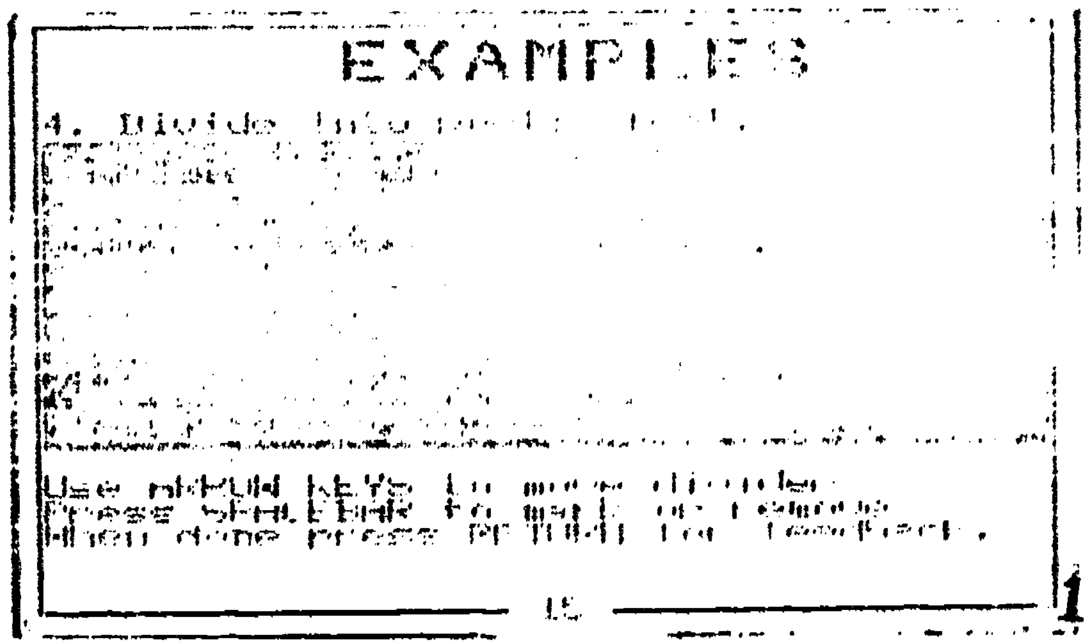
FRAME A



FRAME B

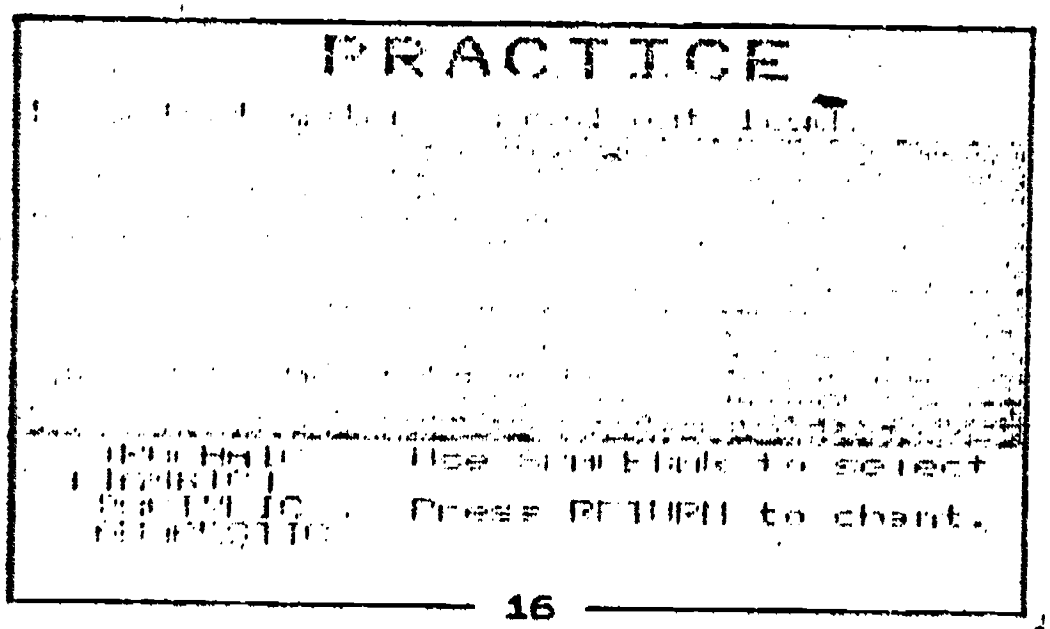


FRAME C



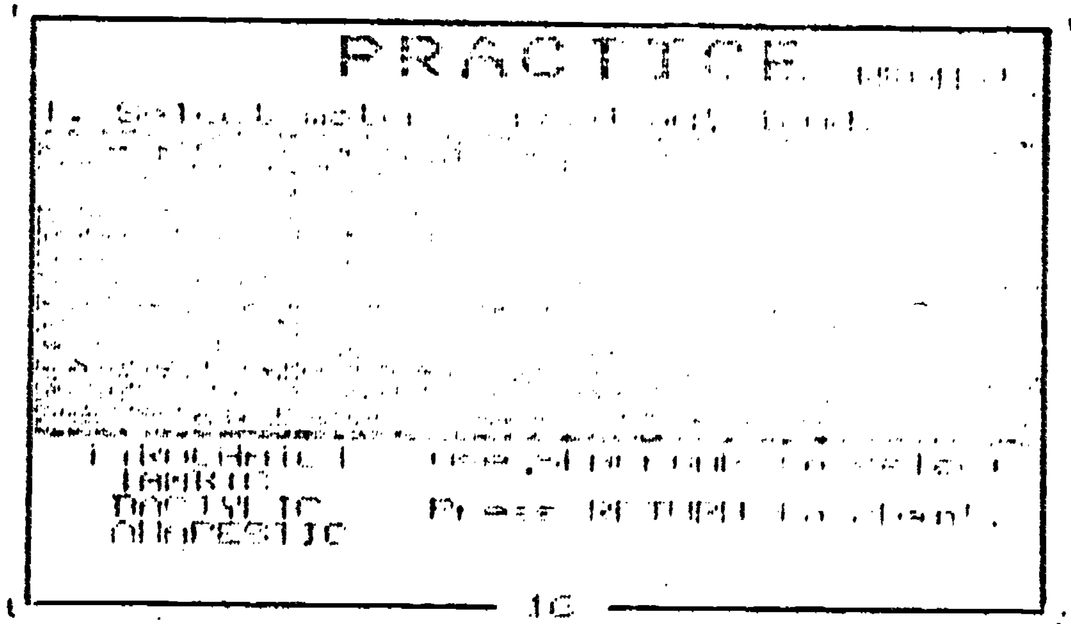
FRAME D

Figure 10 Verse Example with scanning editor - Poetic Meter

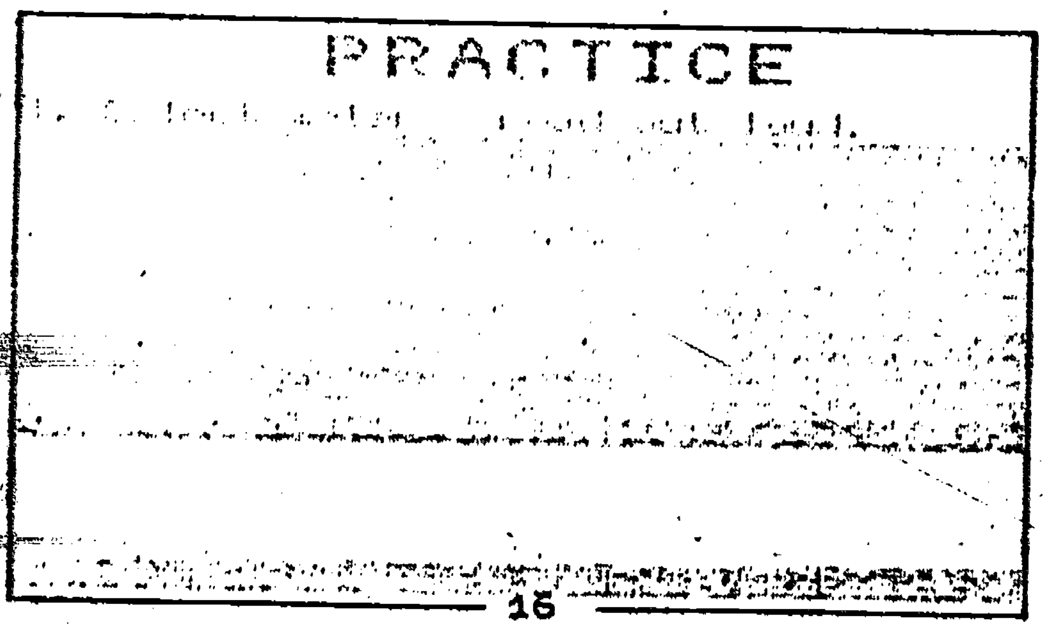


16

FRAME A

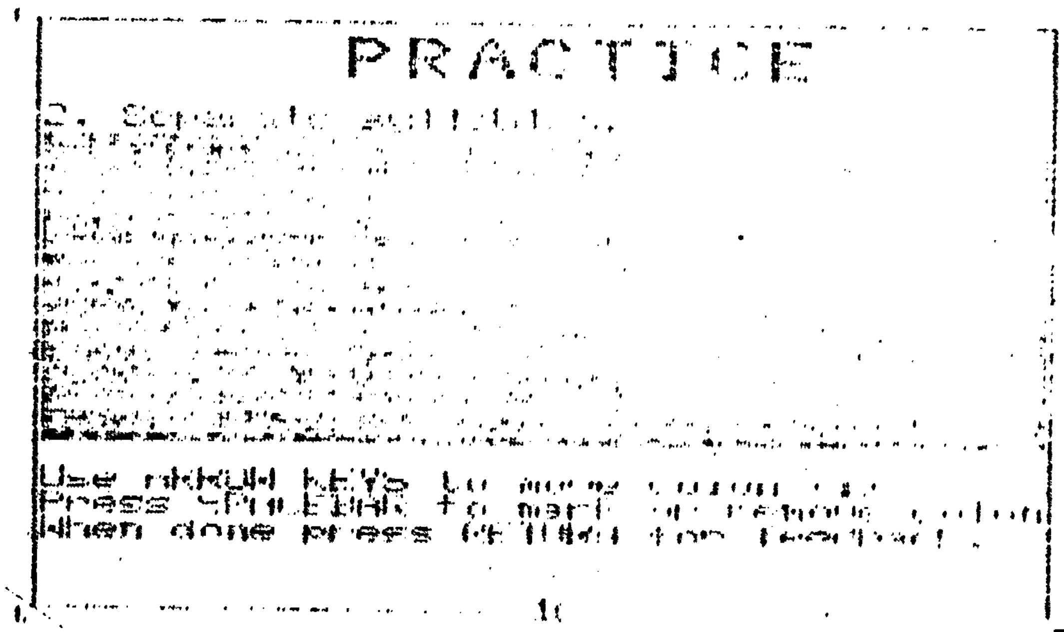


FRAME B



16

FRAME C



FRAME D

Figuer 11 Verse Practice with scanning editor - Poetic Meter

From Domain-Referenced Curriculum Evaluation
To Selection of Educational Microcomputer Software

Wells Hively
President, Hiveley's Choice Publications, Inc.

Much of my past work has been in the field of domain-referenced testing and curriculum evaluation. Some of that work took place in a happy association with the UCLA Center for the Study of Evaluation, which published one of our contributions to this field as the first of its Monograph Series in Evaluation (Hively, Maxwell, Rabehl, Sension, & Lundin, 1973). Those of you who know this work probably will not be surprised at the approach we are now taking to the selection of educational software: compare, contrast, classify, and try to avoid over-generalization.

Currently, we are concerned with evaluating microcomputer programs that can enhance instruction during the period of schooling when it is easiest to consider the curriculum as a whole: preschool through grade 9. We have formed a publishing company to assemble and transmit information about educational microcomputer software to schools. Our purpose is to help school people more easily find what they need and use it more effectively. Specifically, we want to help teachers answer the following questions:

What kinds of programs are currently being developed?

How can we find good ones?

How can we use them effectively in school?

How can we tie them into the basic school curriculum?

We assume that there are many different types of educationally useful microcomputer programs, each with its own practical purpose, each derived from its own theoretical assumptions and each, therefore, requiring its own unique set of evaluative criteria. We also assume that the lesson-plan settings in which teachers use the programs have at least as much influence on their impact as the characteristics of the programs themselves. Consequently, useful evaluation must take account of both the characteristics of the programs and the ways in which they are used.

The terminology used to classify different types of programs has by no means settled down. To make matters worse, the terms often carry evaluative connotations. Currently, outside the military, "CAI" (computer-assisted instruction) is a low-status term. "Drill and practice" is out. "Simulations" are in. "Learning games" are in. "Computer literacy" is definitely in. But all these terms are operationally hazy. It's important to try to clarify the terms we use to classify programs, because the classifications govern our approach to evaluation: programs are relatively easy to compare and evaluate within the same class, but very difficult to compare across classes. Let me give you examples of the different classes of programs we are encountering, and suggest some more precise nomenclature for them.

Types of Programs

A simple and generally useful type of program may be called "Domains of Practice". A good example is a program published by Sunburst Communications called Smart Shopper Marathon. The purpose of the program is to provide practice in rapid arithmetical estimation. The setting is an imaginary supermarket. Each so-called "aisle" represents a different domain of practice. In aisle 1, the student has to rapidly estimate the results of dividing a decimal by a whole number. The student's job is to answer as many problems as possible in a given time, so quickly that detailed calculation is a hindrance: rounding off and estimating is the skill that must be practiced. In aisle 2, the student has to estimate the results of multiplying whole numbers. In aisle 3, subtracting decimals; in aisle 4, comparing fractions; in aisle 5, multiplying whole numbers times decimals.

The problems in each set are generated in random order, and each time you use the program the "aisles" appear in a different order. Therefore, because students are not likely to memorize rote sequences, the program lends itself to repeated practice without boredom. The scores used to judge youngsters' progress are based on a combination of speed and accuracy.

Programs like Smart Shopper Marathon are characterized by items drawn from clearly defined domains of knowledge or skill, a high frequency of opportunities to respond per unit time, and almost total absence of instruction presented by the program itself. The teaching of the constituent skills and the orientation to the problem-solving approach must come from an outside source. The students must obtain guidance in strategies for estimation from their teacher, or from each other, working in a small group. Therefore, programs like this make good vehicles for classroom demonstrations and for small group discussion. They provide external focus and feedback around which classroom activities can be assembled. Increased performance on the domains of practice presented by the computer may become the criterion toward which teacher and students can work together, a welcome change from the teacher's usual job of standard setter and task master.

It is useful to compare programs of the domains-of-practice type with programs of a second type that have historically been called "tutorial". A tutorial program developed by the Minnesota Education Computing Consortium leads up to the geometrical definition of an angle. What the student encounters in programs of this class is a fixed and predetermined sequence of presentations of bits of information interspersed with questions and answers. The term "tutorial" is too broad to clearly denote this type of program. Let's narrow the terminology to "linear tutorial". This is the classic programmed-instruction format which most people associate with the so-called "CAI" that is currently out of fashion. Perhaps one can see the reason why. The frequency of opportunities for students to respond in linear tutorial programs is low in comparison to the rapid-fire opportunities provided in domains of practice. Nearly always, the expository material could be conveyed faster in a book or in a conversation with a teacher or a peer. Perhaps most important, the sequence of "telling and testing" arises totally out of the mind of the designer, with no elbow room given for the idiosyncracies of different learners.

In constructing domains of practice, we are on fairly safe ground because we are creating models of subject matter. The theory and methodology of domain-referenced testing provide a fairly solid foundation for this job. But in linear-tutorial programs we are attempting to model the dynamics of teacher-student interaction without actually allowing dynamic interaction. There is little theory to guide this task, and successful programs of this kind are hard to find. Perhaps artificial-intelligence theory will eventually help us construct truly dynamic tutorial programs of the Socratic or error-analysis type. But good, dynamic tutorial programs have not yet filtered down to the practical level where we encounter them in our survey. For practical purposes, teachers can do much better by putting small groups of children to work on domains-of-practice programs, and letting the teaching arise from class discussion and spontaneous interaction, than by sending individual children off to have learning doled out to them in small droplets by step-by-step, linear-tutorial programs.

An enormously popular third type of program is the education game similar to those seen in video parlors and arcades. Basically, these games are domains-of-practice with several added attractions. This combination may be called extrinsically-motivated practice or extrinsic games. Some of them are a lot of fun.

An example of an extrinsic game is the DLM Company's Alligator Mix. If your answer in the alligator's belly matches the problem in the apple that comes floating in from the left side of the screen, you win by opening the alligator's mouth and swallowing it. If it doesn't, you leave the mouth closed, and it bounces off and spins away. At the beginning, there is just one alligator, at the bottom of the screen, and the apple has to travel a long way, so you have plenty of time to make up your mind. After a string of successes, a new alligator surfaces. The apple doesn't travel as far to get to that alligator, so you have to think faster. When there are four alligators lined up, you really have to hustle. The teacher or student can choose from nine skill levels, which have to do with the velocity of the apple's motion, and three problem ranges which have to do with different domains of practice.

Another example from DLM is called Demolition Division. The tanks all come forward at the same time, shooting at your forts. Your job is to position a 0 underneath the cannon that aims at the most threatening tank. Then change the 0 to a number that corresponds to the questions on the tank, press the space bar, and destroy the tank before it knocks down the wall and destroys your cannon.

What makes these games fun is delicate grading of speeds and levels, freedom to select levels that match entering skill, and richness of alternatives in ways to respond. A whole art and technical literature is growing up in the area, and standard "plot formulas" are rapidly appearing.

Another kind of plot formula for an extrinsic game is demonstrated by Sunburst's Math Mansion. It gets good mileage out of a "dungeons and dragons" theme. The thematic development and the richness of alternatives in Math Mansion trade off against relatively low frequencies of opportunities to respond. We are a long way from knowing, if we ever will know, what are the optimum mixes of such ingredients. But youngsters identify the good examples by their attention and their resultant learning.

A fourth category of program might be called, by way of contrast, intrinsic games. QED Company's Arith-magic program is called "Diffy". The student volunteers a set of four numbers which the computer places at the corner of a square. Then the student goes around the square finding the differences between each pair of numbers. The differences found in the first round then form the corners of a new square for the second round, and the student goes around finding the differences again. This goes on until, eventually, lo and behold, the differences all come out the same. The challenge is to figure out what characteristics of the starting numbers make the differences converge quickly or slowly. The game provides a vehicle for discussion, exploration, and curiosity, and incidentally provides a very high frequency of opportunities for subtraction practice.

Other examples of intrinsic games are Sunburst's _Teasers by Tobbs_ and MECC's _Taxman_ and _Bagels_. Games like these tap into the whole realm of classic puzzles and brain twisters, some of which lend themselves nicely to computer presentation. As usual, the most frequent examples tend to be in the field of mathematics, but there is no reason why they need to be limited to that field.

A fifth promising category of programs is exemplified by two in the Milliken Company's _Edufun_ series: _Golf_ and _The Jar Game_. Let's call them intuition-building programs. In _Golf_ the problem is to direct the ball from the tee to the green by estimating an angle and a distance using a compass rose for reference, and a given unit of distance. If you lead off, you must estimate distance and direction absolutely (in terms of the compass rose and the unit of distance), but if you shoot second, you can correct your direction by adding or subtracting degrees to the course taken by your opponent's ball, and you can correct your opponent's estimate of distance. The game builds up a nice intuitive judgment of angles and directions.

In a similar vein, _The Jar Game_ builds upon the intuitive statistical notion of drawing beads out of a jar. The young student is then shown diagrams of jars containing different proportions of two kinds of candy pieces. The job is to figure out on which jar of candy a randomly-directed fly will land more often.

There are many other potential examples of geometrical and statistical intuition-building activities that computer experiences could enhance. The ease and speed with which the computer can generate these examples is delightful. We have come a long way from the old days of having children estimate the number of raisins in an average slice of raisin bread by taking apart a loaf of bread and counting the raisins in selected slices.

The sixth category is simulation programs. There are so many different kinds of simulations, and they can produce so many different outcomes, that this category will no doubt be subdivided later, but the characteristics that guide subdivision are not yet clear. The MECC Sell Series, built around the famous Sell Lemonade, is an example. The simplest one of the series is called Sell Apples.

When youngsters are turned loose on a program like this, they may learn many different things, depending on the context provided by the teacher. They may learn to read carefully and follow instructions in detail. They may learn to interpret data in tabular form. They may build up an intuition about the relationship beween price and volume of sales. They may learn important habits of record keeping. They may learn to transform data into graphical form and interpret trends. At a deep level, they may learn some important strategies that underlie scientific method, such as choosing extremes of variables and narrowing down to find the maxima and minima. They may even learn something about the cost benefit of seeking truth. None of these things is taught for sure by the program. They depend on the context provided by the teacher and other students. It is particularly obvious, in the case of simulation programs, that validity and usefulness depend as much on the context provided by the teacher and peers as on the programs themselves.

MECC also provides a nice example of the seventh category: information retrieval programs. Nutrition asks you to provide a list of your food intake for one day. Then it gives you a nutritional analysis: how well your day's food intake represents the four basic food groups, how your numbers of calories provided by fat, carbohydrate, and protein compare to the recommended number of calories for a person of your age and stature, and how your intake of iron, calcium, vitamin A, and vitamin C compares to the recommended daily requirements. This is what the computer does best, and its role in this kind of instruction is distinctive. MECC's Nutrition program does not provide a means of accessing or adding to the nutritional data base, but one can easily imagine programs to which teachers and students might add information for foods not currently included, or ask for other kinds of analyses; a nice meeting ground between specific subject matter and computer literacy.

Nutrition is a miniature data base, and elsewhere a wonderful array of useful data bases is becoming available to teachers and students. Compuserve, for example, is a service that enables computer owners to obtain information from many data bases at night--airline schedules, weather reports, and so on. The potential of data bases such as these as vehicles for instruction is tremendous. Answering questions that come up in class by accessing a nutritional data base overnight would be a considerable step up from writing letters to the Department of Health.

The eighth and last category is such a large and heterogenous category that it, too, will no dcubt soon be subdivided. For now, let us call it "tools and displays". In this category are all the programs that perform helpful calculations, the programs that process words, programs that display graphs of changes in phenomena detected by sensing devices like thermometers, and programs like the famous Logo that offer environments with important properties to be explored. The educational utility of these programs is limited only by imagination and experience. The following is just one example.

A program produced by Spinnaker Software called Delta Drawing is a kind of baby Logo. The commands are easy to understand, and a small child can start making interesting pictures almost immediately. We start with an arrow, move it forward by pressing the D key, and turn it to the right by pressing the R key. We change the color of the line by pressing the C key and then a number corresponding to the color we want. We move it forward again and change its direction and color. We store all the preceeeding steps in a sub-routine which can be repeated. We repeat the sub-routine to generate a kind of rose window. We may fill the spaces on the screen by choosing a color and then pressing Control F. The computer keeps track of all t steps as a string of symbols, and we can switch to text mode from the graphics mode and examine the string, operate on it, and go back to graphics to see the results.

Programs like Delta Drawing offer nice opportunities to explore symmetries and artistic effects. For example, when a line passes beyond th- border of the screen, we have a choice of having it wrap around and appear from the opposite border or having it bounce off at an equal angle. The line bounces and bounces again like a billiard ball. It continues bouncing to generate a symmetrical pattern.

It is also possible for a repeated figure to wrap around and then bounce to produce a complicated effect. It goes on bouncing and creates an interesting artistic result made up of a combination of expected and surprising features. There is considerable potential in programs like these in the hands of teachers with sensitivity to some of the relationships between art and mathematics.

The overwhelming impression one gets from watching children work with all the foregoing different types of programs--ranging from open-ended environments, like Delta Drawing, to practice sequences like Smart Shopper Marathon--is that their effectiveness depends at least as much on the classroom context in which they are used as on the properties of the programs themselves. Properties of programs which are drawbacks in one context may be benefits in another, and exciting uses may be totally unanticipated by the people who developed the programs.

A Curriculum Guide

With the foregoing review of types of programs as background, now let me tell you about the product of our work: a curriculum guide for grades 0-9 called Hively's Choice. The target audience is what you might think of as "second wave" educators--not the original enthusiasts, but the experienced and thoughtful mainstream teachers on whom any successful educational innovation depends.

Several characteristics distinguish the guide from other efforts to help teachers evaluate and choose software. First, the guide only contains software that has been found to be particularly outstanding in quality and ease of use. Second, the guide is designed in such a way as to make it as easy as possible for teachers to connect the recommended software to curricula and lesson plans. This is done in several ways. The user may begin by looking at a chart showing where each of the programs fits into general subject matter areas and the grade levels over which it is likely to be useful. Next, the user turns to a set of quick descriptions, organized by subject matter, within grade levels, and arranged so that one can look through them very rapidly so as to maximize chances of discovering unexpected connections to upcoming lesson plans.

The reader who finds something of interest by perusing the quick descriptions may turn to a detailed description of that program. There, the goal is to describe the program in enough detail that one can intelligently decide whether it would really be useful and exactly how it would relate to ongoing curriculum.

A subsection of the detailed description called "Curriculum Connections" includes words and phrases that can be used as cross references to scopes and sequences. "Objectives" briefly describes the kinds of learning which may be enhanced by the program, and a section called "Instructional Examples" gives recommendations about how best to utilize the program in classroom discussion, small group or individual work. "Estimated Engagement Time" helps teachers plan how much time to allow for work on the program by the whole class, small groups, or individuals.

Also in the detailed description section one may find the technical information about the program, the hardware it requires, and the name, address and telephone number of its producer. The rest of the book consists of cross indices by subject matter and topic, and an alphabetical listing to facilitate the location of specific programs.

Our goal is to find rich, engaging and easily usable programs that have solid connections to all the areas of the basic curriculum, preschool through grade 9, and that take all the different forms described in the earlier part of this paper. If you imagine the curriculum as a matrix of subject matter areas by grade levels, some of the cells in the matrix are already getting crowded while others are virtually empty. Over time, our goal is to weed out programs in the crowded cells so as to include only a selection of the most useful and interesting ones, and to seek entries in the empty cells. Each year the guide book will be revised, following the analogy of a European travel guide. Like a travel guide, each edition will be cumulative and self-contained.

Organizationally, this work is done by a small, carefully selected group of contributing editors, who work in schools and work very closely with teachers in training. These editors are chosen to represent areas of the country where thoughtful and interesting work with microcomputers is going on. In their daily work with teachers, the contributing editors keep an eve out for outstanding programs and interesting ways of using them. They forward their reviews to a small central editorial staff that produces the book.

In meetings of this editorial staff, we work to explicate the bases for our selections, to clarify categories of programs and the evaluative criteria applicable to each, to organize observations about effective classroom use of various types of programs, and to identify useful sequences and combinations of programs. This work aims to create a dialogue between good theory from the technical literature and careful observations of classroom use. From these, we are developing, year by year, a progressively more useful, readable, and balanced curriculum guide--one which can contribute both substance and diversity to the curriculum for preschool through the first nine grades.

Reference

Hively, W., Maxwell, G., Rabehl, G., Sension. D., & Lundin, S. (1973). Domain-referenced curriculum evaluation. Los Angeles: UCLA, Center for the Study of Evaluation

Comprehensive Evaluation of Computer Courseware: Getting Back to the Basics

Kenneth A. Sirotnik

Laboratory in School and Community Education
Graduate School of Education
University of California, Los Angeles

It should not be necessary to write this report. Computer courseware is simply part of a schooling curriculum. Evaluating courseware should be a natural part of curriculum evaluation. The principles and practices of curriculum evaluation have been developed and refined for nearly four decades ever since the seminal work of Ralph Tyler in the late forties. Nothing terribly new will be found in this report that was not either explicit in or implied by Tyler (1947) or that has not appeared in modernized and expanded work on curriculum theory (e.g., Goodlad, et al., 1979).

Yet it seems that we are rather easily seduced by the promise of educational innovations--microcomputers being the most recent example. And, in our attempt to evaluate the promise, our thoughts turn more to surface-level, technological issues rather than to deeper meanings that we have known for some time to constitute the basic questions of curriculum and instruction.

Consider the advent of educational television; we became preoccupied with questions like: How many? Where should they be located? What are the trade-offs between black-and-white versus color? What is(are) the optimal: number of minutes per program; ratio of talk to action; ratio of cartoon to real-life content; tonal qualities for attention-getting and maintenance? When learning machines came along, we asked: How many? Where should they be located? How small can they be? How much text should there be per frame? What are the trade-offs between linear versus branching programs? What are appropriate forms of reinforcement for correct answers and feedback for errors? What is the optimal ratio of active and passive student interface with the machine? Is it best suited for drill and practice versus higher level thinking and learning?

Clearly there is a generic set of questions here as can be inferred from the ones now being asked about computers and courseware: How many? Where should they be located? Relative advantages of color and graphics? Amount of text per screen? Appropriate use of sound and voice simulation? Appropriate forms of corrective feedback and reinforcement? Optimal ratios of student versus program control? Teaching facts and comprehension versus higher-order, problem-solving modes? These questions, and others like them, raise important issues indeed. For ease of discussion here, I will include them all in a general category labelled technology. Perhaps the single, most important representative of this category is the following question: What does the technology do that could not have been done at least as efficiently and effectively with ordinarily available learning resources (teachers, peers, paper and pencil, books, manipulatives, etc)? Many documents are now available that suggest evaluative criteria for courseware technology and I will not belabor the issue any further here.[1]

But, as hinted at above, there are two other categories of questions that in many ways transcend the particulars of any given technology. Again for purposes of discussion, I will label these the categories of curriculum and assimilation. By assimilation, I mean the process by which schools innovate; that is, in this case, the ways in which a new technology becomes part of the everyday work life of teachers and students. Unfortunately, the track record of districts and schools in innovation and change generally, and the assimilation of technology specifically, is not good. With regard to microcomputers and accompanying software, the cynic might well ask:

Will these testimonials to human genius take their place on shelves along side their counterparts of past eras (teaching machines, educational TV, and the like) only to collect the dust of innovative non-events?

The optimistic evaluator must ask: How has the learning environment (human and material) been modified to receive and exploit (in the best sense) the full potential of computer courseware? To pursue this issue further is beyond the scope of this report. Obviously, from an evaluative standpoint, it would be foolish to hold any particular piece of courseware accountable for the larger issue of assimilation. However, since the viability of computer courseware rests upon the adequate resolution of this issue, it deserves special mention in any evaluation framework.[2] I would now like to turn to the main evaluative thrust of the present paper--curriculum.

Hopefully in what follows, the emerging evaluative questions, should they strike you as old and obvious, will strike you as essential and preemptive of those that might be leveled at the technology (e.g., micros) and its by-products (e.g., courseware). However, given the immediate purpose of this report, these issues will be discussed in the context of computer courseware developed for classroom use.

Let's begin with content and what it is not: CONTENT IS NOT THE CURRICULUM. A comprehensive view of curriculum and any aspect of it (e.g., computer courseware) requires explicit acknowledgement and consideration of all relevant elements of the teaching and learning experience.

The fact that content per se is part of but not the whole of curriculum was implicit in Tyler's (1947) rationale. Since then, various attempts have been made to sort out the facets of curriculum as a multidimensional construct. One that has seen a good deal of theoretical and practical development is Goodlad's (1979) notion of curricular commonplaces. This idea has enjoyed considerable success both in framing curriculum inquiry (Goodlad, Klein, and Tye, 1979) and generating relevant data for the study and assessment of schooling (Klein, Tye and Wright, 1979; Sirotnik and Oakes, 1981).

The list of curriculum facets that can be seen as commonplace to all organizations of teaching and learning experiences includes at least the following:

1. Goals/Objectives--statements of intended teaching and learning specific enough to convey at least the relevant content and expected behaviors.

2. Content--substantive strands and topics comprising the "stuff" of teaching and learning.

3. Strategies--instructional methods or processes designed to promote teaching and learning, e.g., use of open-ended questions, group discussions, lecture, deductive/inductive approaches, etc.

4. Activities--events, tasks, etc. designed to engage teachers and learners, e.g., reading, writing, listening, practicing, role playing/simulating, etc.

5. People--human resources available to facilitate teaching and learning, e.g., teachers, aides, student peers, etc.

6. Materials--physical resources designed to facilitate teaching and learning, e.g., pencils, paper, books, learning kits, manipulatives, calculators, computers, television, etc.

7. Grouping--ways in which human resources are organized for teaching and learning, e.g., total class, individual seat work, cooperative learning groups, etc.

8. Time--allocation and use of time in teaching and learning.

9. Space--ways in which classroom areas are organized for teaching and learning.

10. Assessment--determining, collecting and interpreting information for describing and judging the effectiveness of the teaching-learning process and for facilitating decision-making and action-taking towards the improvement of that process.[3]

But these commonplaces alone do not provide an adequate framework for making curricular judgements. They suggest appropriate types of descriptive information but lack the bases for evaluation. The commonplaces answer "What is?" questions, but do not automatically address "Why and what ought to be?" questions. The needed evaluative screens--what I will call here educational values and beliefs--permeate all curriculum inquiry. It is simply a question of making them explicit along with, hopefully, some working consensus among those concerned about why these "oughts" are important.[4]

I have a list of educational values and beliefs that I will share with you, more for the purpose of illustration than indoctrination. Many "oughts" on this list will be familiar to you since they appear in most formal curriculum documents at state and local levels. Most are implicit in Tyler's (1947) discussion of the sources and criteria for, and the organization and evaluation of, learning experiences. Many are implicit in the Goodlad, Klein and Tye (1979) dimension of "qualitative factors" which they suggested as evaluative screens for curriculum commonplaces. All are compatible with my own orientations and experiences as a student, teacher, and parent, and as an educational researcher in collaboration with my colleagues on A Study of Schooling (see Goodlad, 1983) and, more recently, on projects of the Laboratory in School and Community Education.[5]

Consider, then, the following list of keywords, each intended to represent a constellation of educational values and beliefs:

1. Equity--equal access to the curriculum (content, teaching practices, time, etc.) regardless of race, sex, religion, etc. or any correlates thereof (e.g., social economic status).

2. Experience--building upon concrete, real-life events, feelings, and meanings in the empirical world of teachers and students.[6]

3. <u>Critical Thinking/Problem Solving</u>--going beyond the necessary facts and comprehension levels of cognitive processes, questioning knowledge, and using higher order processes such as analyzing, synthesizing, proving, applying, abstracting, and evaluating.

4. <u>Discovery/Creativity</u>--freedom to explore knowledge, think divergently, invent, imagine, and so forth.

5. <u>Proactivity</u>--deliberate involvement of students in their own learning such that they become active, non-passive, and non-reactive decision-makers.

6. <u>Integration</u>--treating knowledge "ecologically," not as discrete, unrelated bits of information, but as parts contributing to a whole.

7. <u>Variety</u>--deliberate use of different instructicnal activities, materials, grouping techniques, etc. in contrast to an over reliance on only one teaching-learning configuration (e.g., teacher lecturing to the total class).

8. <u>Individual Variability</u>--recognizing individual differences in ability, learning styles, attitudes, interests, etc. as assets rather than liabilities and adjusting/adapting/modifying curricular elements to accommodate these differences.

9. <u>Socialization</u>--humanizing knowledge through exploring why and how it is <u>not</u> independent of its sociocultural and political context.

10. <u>Personalization</u>--humanizing knowledge through exploring personal meanings, sentiments, interests, and future aspirations.

Clearly, these categories are not mutually exclusive. Some might even be combined with no loss to the usefulness of the framework; for example, critical thinking, problem-solving, discovery and creativity might all be combined into a general category labelled <u>inquiry</u>. Others need to remain separate yet clearly interact in fundamental ways (e.g., considerations of equity render ability tracking indefensible as a school policy for dealing with individual variability).

Nevertheless, whether these or some others, some list of values/beliefs must be made explicit as a frame of reference for evaluating curriculum. A convenient way to map this evaluative task is to form the matrix of questions that naturally emerges by crossing an educational values and beliefs list with the list of curriculum commonplaces. (See Figure 1.) Each cell represents the obvious set of questions that are generated by the interaction of the value/belief represented in given row and the curriculum commonplace of a given column. For example, following through with a commitment to dealing effectively with row eight of the matrix requires asking questions like: Do the topics, activities and teaching strategies accommodate the different learning styles, abilities, etc. of the students? Are appropriate material and human resources available and used for accommodating these differences? Are students taught en masse, at the same time, and in the same place or are allowances made depending upon individual differences (using small groups, variable pacing, learning centers, etc.)? Is testing primarily summative for the purposes of uniform grading or formative for the purposes of diagnosis and facilitating individual learning progress?

The point of imposing this kind of comprehensive, evaluative screen on some computer courseware program should be obvious. Courseware does not exist independently of curriculum. It contains and/or addresses--or should address--all curriculum commonplaces. As consumers, we must evaluate the courseware accordingly. We must resist the temptation to be sold only by the flashy novelty of a technological invention and demand some understanding and judgement regarding how the invention fits into the desired curricular scheme of things.

The values and beliefs in Figure 1, for example, lead one to ask questions like: Is the courseware biased with respect to one or more demographically defined student subgroups? Can the courseware tap into real-life student experiences? Does the courseware address higher-level cognitive skills and processes? Is discovery learning, the exploration of concepts, and inventing new concepts encouraged by the courseware? Is it clear how the courseware is an integral part of the larger curriculum? Are individual differences in ability, learnings styles and so forth accommodated by the courseware? Is it clear how the courseware is but one instructional vehicle among many, i.e., how it can be interfaced with teachers, peers, ordinary materials (pencil, paper, manipulatives, etc.), learning center activities, etc.? Is the learner treated by the courseware as a passive recipient of knowledge or as an actively engaged learner and decision-maker?

These kinds of questions and/or others like them must be addressed when evaluating courseware or any significant aspect of curriculum. In so doing, it mitigates against simplistic evaluations of courseware like: "Oh, that's just a drill and practice worksheet or textbook simulator" or "Look how wonderfully this program simulates human intelligence." It is almost as though the labels "drill and practice" and "artificial intelligence" carry with them self-evident properties of "bad" and "good" respectively. Clearly, however, drill and practice courseware and programs such as LOGO can be either useful or useless depending upon how their use addresses the issues and questions suggested by a matrix such as the one in Figure 1.

In summary, it should be understood that I am <u>not</u> suggesting that any particular piece of courseware be held accountable <u>in and of itself</u> for each and every cell of a values/beliefs-by-commonplace matrix. Rather, the suggestion is that curriculum must be held accountable in this way and, therefore, so must educational software. To put it another way, checklist-type evaluations of courseware stripped of their instructional context will be insufficient to guide selection. Certainly the information collected by these checklist evaluation techniques is useful, but particularly as it is brought to bear upon, and revised in accordance with, the intended classroom curriculum. Ultimately, thoughtful consideration of the tough questions of curriculum inquiry must be imposed upon courseware as it is _used_ in the specific educational setting.

Footnotes

1. See, for example, the checklists in reports by Edwards (1984), Marshall (1984), Merrill (1983), Hively (1983), and Van Buskirk (1983); see also the guides published by (1) the California Library Media Consortium for Classroom Evaluation of Microcomputer Courseware, (2) The Computing Teacher, (3) the Educational Products and Information Exchange, and (4) MicroSIFT (The Computer Technology Program, Northwest Regional Educational Laboratory).

2. Many readings currently exist in the area of school innovation and change, generally, and the assimilation of technology, specifically. Examples are: Oettinger (1969); Sarason (1971); Goodlad (1975); Heckman, Oakes and Sirotnik (1983); Mayer (1983); Oakes and Schneider (1984).

3. All these curriculum commonplaces deserve considerable elaboration but this is beyond the scope of the report. Nevertheless, assessment deserves special mention because of the popular tendency to equate it only with student achievement testing. Certainly some information is conveyed through an accounting of items answered correctly, especially on a test designed to measure course objectives (i.e., criterion-referenced testing). But much more is possible and desirable, for example:

   o exploring error patterns as in answer-until-correct formats for multiple-choice items (Wilcox, 1984).

   o developing testing strategies commensurate with various learning styles suggested by recent work in cognitive psychology (e.g., Glaser, 1981; Mayer, 1983).

   o Thinking of assessment as formative (vs. summative) and incorporating routines being recently suggested in the areas of diagnostic testing (Ekwall and Shanker, 1976; Thomas, 1981).

   o expanding the domains of testing to include attitudes, feelings, impressions, etc. as they relate to the intended curriculum.

   o expanding response formats beyond the closed ended item, e.g., open-ended, short-answer and/or essay-type response (perhaps printed out for on-the-spot or later teacher analysis and feedback).

   The creative assessment of teaching and learning has been possible for some time; technology can make it more feasible and efficient. Why not make use of it, then, in more ways than just simple question-answer formats?

4. The process, of course, is not so simple. The basis for achieving a "working consensus" has been a matter of considerable philosophical debate in the arena of epistemology, i.e., what constitutes knowledge and the means whereby it is obtained. The position advocated here is multi-paradigmatic. It embraces both quantitative (traditional research using experimental and correlational designs and statistical analyses) and qualitative (naturalistic research using ethnographic, case study, and observational techniques and interpretation) methodologies, so long as a critical perspective is maintained. By this I mean a rigorous and sustained dialogue that addresses such questions as: What goes on in the name of curriculum? How did it come to be that way? Whose interests are being served by the way it is organized? Is this the way we want it to be? We have used the term critical inquiry to describe this multi-paradigmatic perspective. The interested reader is referred to Sirotnik and Oakes (1983).

5. Two projects particularly stand out in this regard: The Curriculum, Computers and Collaboration Project (see the "criteria-by-commonplace" matrix in THE PARTNERSHIP Newsletter, 1 (3), p.7) and the curriculum inquiry task force of THE PARNTERSHIP.

6. This value/belief is directly traceable to Dewey (e.g., 1938) and the progressive education movement. See, also, the more recent extrapolations of these ideas in, for example, computer education and Papert's (1980) "microworlds" and mathematics curriculum inquiry and Romberg's (1983) "story shells."

References

Dewey, J. Experience and education. New York: Collier, 1938.

Edwards, L. Three steps to courseware selection. School Microcomputing Bulletin, 1984, 3, 185-186.

Ekwall, E. E. & Shanker, J. L. Diagnosis and remediation of the disabled reader. Boston: Allyn and Bacon, Inc., 1976.

Glaser, R. The future of testing: A research agenda for cognitive psychology and psychometrics. American Psychologist, 1981, 36, 923-936.

Goodlad, J. I. The dynamics of educational change. New York: McGraw-Hill, 1975.

Goodlad, J. I. & Associates. Curriculum inquiry: The study of curriculum practice. New York: McGraw-Hill, 1979.

Goodlad, J. I., Klein, M. F. & Tye, K. A. The domains of curriculum and their study. In J. I. Goodlad, et al., Curriculum inquiry. New York: McGraw Graw-Hill, 1979.

Goodlad, J. I. A place called school: Prospects for the future. McGraw-Hill, 1983.

Heckman, P. E., Oakes, J. & Sirotnik, K. A. Expanding the concepts of renewal and change. Educational Leadership, 1983, 40, 26-32.

Hively, W. Hively's choice. Elizabethtown, PA: Continental Press, 1983.

Klein, M. F., Tye, K. A. & Wright, J. E. A Study of Schooling: Curriculum. Phi Delta Kappan, 1979, 61, 244-248.

Marshall, W. K. Assessing educational software. Paper presented at the conference of the American Educational Research Association, 1984.

Mayer, R. E. Human cognition and the use of new technologies. In E. L. Baker & J. L Herman (Eds.) The role of new technologies in schools: Collected papers. Los Angeles: Center for the Study of Evaluation, UCLA, 1983.

Merrill, D. Instructional software design principles. In E. L. Baker & J. L. Herman (Eds.) The role of new technologies in schools: Collected papers. Los Angeles: Center for the Study of Evaluation, UCLA, 1983.

Oakes, J. & Schneider, M. Computers in the classroom: Another case of the more things change the more they stay the same. Los Angeles: Laboratory in School and Community Education, UCLA, 1984.

Oettinger, A. G. Run, computer, run: The mythology of educational innovation. Cambridge: Harvard University Press, 1969.

Papert, S. Mindstorms: Children, computers, and powerful ideas. New York: Basic Books, 1980.

Romberg, T. A. A common curriculum for mathematics. In G.D. Fenstermacher & J. I. Goodlad (Eds.) Individual differences and the common curriculum. Chicago: University of Chicago Press, 1983.

Sarason, S. B. The culture of the school and the problem of change. Boston: Allyn and Bacon, 1971 (1st edition) and 1982 (revised edition).

Sirotnik, K. A. & Oakes, J. A contextual appraisal system for schools: Medicine or madness? Educational Leadership, 1981, 39, 164-173.

Sirotnik, K. A. & Oakes, J. Critical inquiry and school renewal: A liberation of method within a critical theoretical perspective. Los Angeles: Laboratory in School and Community Education (Occasional Paper No. 4), UCLA, 1983.

Thomas, R. M. A model of diagnostic evaluation. In A. Lewy & D. Nevo (Eds.) Evaluation roles in education. London: Gordon & Breach, 1981.

Tyler, R. W. Basic principles of curriculum and instruction. Chicago: University of Chicago Press, 1949.

Van Buskirk, T. Hands-on beginner's manuals. T V Ontario, Ontario Educational Communications Authority, 1983.

Wilcox, R. R. Measuring mental abilities with latent state models. Journal of Mathematical and Management Sciences, 1984, in press.

CURRICUM COMMONPLACES

EDUCATIONAL VALUES AND BELIEFS	GOALS and OBJECTIVES	CONTENT	STRATEGIES	ACTIVITIES	PEOPLE	MATERIALS	GROUPING	TIME	SPACE	ASSESSMENT
EQUITY										
EXPERIENCE										
CRITICAL THINKING/ PROBLEM SOLVING										
DISCOVERY/ CREATIVITY										
PROACTIVITY										
INTEGRATION										
VARIETY										
INDIVIDUAL VARIABILITY										
SOCIALIZATION										
PERSONALIZATION										

Figure 1

An Evaluative Matrix
for Describing and Judging
Curriculum

Computers in the Classroom:
Another Case of the More Things Change
The More They Stay the Same?

Jeannie Oakes
Laboratory in School and Community Education
UCLA

Mark Schneider
Norwalk High School
Norwalk-La Mirada Unified School District

1983

DRAFT

Do not cite or circulate without permission of authors

Computers in the Classroom:
Another case of the More Things Change, the More They Stay the Same?

Consider the following--computers are changing the fundamental character of schooling; the microchip will dramatically alter the process of education; teaching and learning will be transformed by the power of digital technology. Familiar statements? Certainly. Widely held views? Of course. Supportable assertions? Well, a great deal of evidence of a micro-boom in school is all around us. But does all this activity mean that we are in the midst of an educational as well as technological revolution spearheaded by the microcomputer? True, we have as yet only glimpsed at what computers might be able to do educationally, but we've had some long hard looks at booms and trends. We've seen booms burst; and we've seen trends nearly always end. If we judge from the computerized curricula we now have available, an educational revolution does not seem to be at hand. If the current use of computers in classrooms is taken to mean that we are experiencing a teaching and learning revolution, it is certainly one of micro proportions. If the character of schooling is changing, it is doing so only in the most superficial ways. If the microchip is altering the process of education, it is as yet only to the slightest degree. The potential for change may be enormous, but if we look closely at what actually happens now in computerized schooling it bears a striking similarity to traditional classroom practice. The infusion of computers into schools seems to be yet another case of the more things change, the more they stay the same.

How is it possible that sameness is the result of all the computer activity around us? Both rhetorical and physical signs of change abound! We know, for example, that computers are everywhere in schools. Early reports from a recent survey of microcomputer use in schools by John Hopkins University indicate that, as of January of 1983, 53% of all schools (public and private) had at least one computer for instructing students. Further, the study found a rapidly accelerating rate of computer acquisition by schools during the last 2½ years. Secondary schools are leading in the computer-acquisition race with over 85% of all senior highs having computers for instruction. Amazingly, the number of secondary schools with five or more micros doubled between June of 1982 and last January, and they now constitute 40% of all secondary schools. (Becker, 1983).

Computer literacy courses have appeared everywhere as part of the curriculum and computer literacy was named as one of the <u>five new basics</u> that should be required for high school graduation by the National Commission on Excellence in Education in its recent report <u>A Nation at Risk</u> (1983). Teacher inservice courses in the use of microcomputers proliferate in the most prestigious university schools of education, in the not-so-prestigious college weekend extension programs, and in the blatantly commercial storefronts and offices of hardware manufacturers and sellers. While these courses range widely in both their style and substance, "Don't be left behind" appears to be their most salient message. As the world is being revolutionized by computers, the future of schools and teachers, it seems, will be digital as well.

In a more subtle, but equally potent form, the message is sounded in the onslaught of hardware and software and salesmen who wax eloquently, if not always intelligibly, about bits, bytes, roms, rams, and the ideal number of K. At a recent conference attended by over 3,000 computer-using educators, the traditional separation between scholarly and commercial presentations was blurred, and few seemed to question the mixing of the two. Also noticed was a curiously hard version of soft-sell. "Yes, there is little in the way of truly useful software on the market," those attending the conference were told. "No, computer literacy is not the way to go," the experts said. "Why get a machine to do what a human can do better, more sensitively, more cost-effectively?" all agreed. "I can't spend my budget; there's so little worth buying," was a frequently heard complaint. Yet there was no doubt amidst the sharing, showing, and selling that software budgets would be spent and machines would be bought as soon as the manufacturers could get the newest versions to work properly.

How is it possible that computerized education could even begin to resemble, let alone replace, human-to-human teaching and learning. "Not to worry," teachers are constantly reassured. Computers are not the smart machines we sometimes give them credit for being; computers only know what humans teach them. Computers will never provide the multiplicity of modes and responses that a sensitive human teacher has at his or her fingertips. Computers will never be able to respond appropriately to the divergence in students' creative output, to weigh carefully the distinctions in differences of opinion, to interpret carefully in questions of values, or to ferret out a complicated thought process. Computers will never communicate the joy of discovery and the pleasure of watching and helping someone learn. Teachers will always be required for those subtle and complex interactions thought to be the heart of the teaching/learning process.

But are these human interactions, in fact, at the center of teaching and learning in classrooms? Those of us who analyzed the data about the teaching and learning process in the 1000 classrooms that constituted the sample for A Study of Schooling found something quite different from this ideal picture of classroom instruction (Goodlad, 1983). Teachers who employed a wide variety of instructional strategies or arranged for students to experience a variety of learning modes were extremely rare. Rather, with an amazing consistency, teachers across the grades and in nearly every subject area relied on two dominant instructional configurations--1) lecturing or demonstrating to the whole class, and 2) having students work alone using texts or workbooks or worksheets.

Teaching was almost exclusively the presentation of information. Learning was nearly always seen as the passive intake of information or as practice. Within these two classroom configurations, teachers out-talked their students by a ratio of nearly three to one. Importantly, most of this teacher talk consisted of telling--the presentation of information. Less than 6% was in the form of questions. During the small amount of questioning that took place, less than one sixth of the questions were open-ended, requiring students to respond in complex ways--to evaluate, to analyze, to react. Most questions required answers like "yes" or "no" or specific facts like "Columbus" or "1492" (Sirotnik, 1983).

Further, we found in these classrooms an emotional climate that can best be described as flat. Little in the way of warmth and enthusiasm, encouragement or praise was expressed by teachers. Nor was there evidence of much eagerness, curiosity or overtly positive responses by students. Happily, overtly negative behavior was noticeably absent as well. But it is disconcerting, at best, to think of 95% of teaching and learning taking place in an environment so neutral that it is hard to believe that anyone cares very much about what is going on.

For the most part teaching and learning activities were traditional and passive--teachers lecturing and students listening, or students working alone on written assignments. Rarely were more active learning modes found. Chances were less than 8% that students in these classrooms would be involved in discussion, simulation, role playing or demonstration. Students worked cooperatively only 10% of the time.

Passive instruction is not news. Studies as far back as the turn-of-the-century report this familiar classroom scenario (Stevens, 1912; Hoetker and Ahlbrand, 1969). But the evidence certainly does cast doubt on the important and central role in classrooms of the kinds of subtle interactions we say we value in teaching and presume computers cannot duplicate. These uniquely human qualities may be more instructional myth than reality.

Enter the computer! A device which allows students of varying abilities to cover various materials at varying rates. The key word here is vary (we suppose individualize hints too much of the "flaky sixties"). Let's look at what is being varied and for what reasons. Does this variety benefit the student by accommodating varied learning styles and encouraging more active modes of learning, or does it benefit the teacher by allowing greater ease in following traditional modes of teaching? A look at the most common types of computer-based materials will help us answer this question.

Drill and practice. Drill and practice is the predominant mode of computerized instruction in use today. Its roots can be traced to the teaching machines heralded in the 50's and trashed in the 60's. Students may be asked to answer math problems, choose the correct spelling for a word, or fill in holes in sentences. The key seems to be the ease with which an answer can be marked right or wrong. Any objective knowledge that can be memorized, spit back, and easily judged for correctness is prime material for a drill and practice program. Not only are companies beating down classroom doors to sell software drills on every conceivable subject, but teachers who have developed their own simple drill programs are joining the commercial competition as well, either on their own or through software houses. Advertisers are touting the phrase classroom tested, as if this label has a bearing on the appropriateness of the programs. Recently, classroom tested drill and practice courseware has been enhanced with the addition of limited authoring capabilities which allow teachers to type in their own lists of questions and answers so that the materials can be more easily tailored to a particular course. This makes the programs a bit closer to what those teachers have been doing already. Some revolution!

Who benefits from these drill and practice programs? Does the student learn more or better with the questions on a screen instead of in a workbook? Although students may be more motivated when they see their name on TV, for how long will this fascination last in today's world of video games and space technology? It may also be argued that the learning is more individualized because students can study different lists, but is a computer really necessary to accomplish this task? It definitely is easier for the teacher with the computer, as the record-keeping capabilities of many courseware packages free the teacher from such chores. But does this alter the educational process, or is it just more of the same in a new package?

Tutorials. In the tutorial instructional mode, the computer lectures to the student on topics ranging from spelling rules to nuclear fusion. If drill and practice is linked to electronic workbooks, tutorials may be compared to electronic textbooks. A typical program leads a student through the material being presented, the only variable being speed based on individual reading rate. This approach may be worthwhile if the material is graphically presented on the screen in a way superior to the chalkboard, film, video, books, etc. More elaborate tutorials allow students to repeat sections they are not sure of, but few programs help students decide when repetition is desirable. The only variables are those that relate to how fast the student reads, and how quickly the material is understood. Everyone goes through the same material presented in the same way. Student conscientiousness and study habits probably account for more program variation than student learning.

Who benefits from these tutorials? Certainly not students whose reading level may be below the comprehension level for the material being presented. Furthermore, it is doubtful that other students will learn more than they could from a live presentation which might encourage more active learning modes such as discussion, questions, and interaction with other students. On the other hand, the teacher does not need to prepare each demonstration or worry about repeating the demonstration for absent students. Even slower students are provided for: "If they do not get it the first time, let them view it again, the computer is patient."

Again, the tutorial programs, like the drill and practice, seem to make it easier for teachers to retain status-quo teaching strategies. The primary modes of teaching have not really changed--only the labels applied to them. Teacher lecture has given way to a slicker computer lecture, and workbook drill has been replaced (or in many cases augmented) by electronic drill. And what of the flat, uncaring tone present in the traditional classrooms? It is folly to suppose that a computer can express more in the way of warmth and enthusiasm, encouragement or praise than a human teacher, though some programs come in gift boxes and psychedelic ribbons few humans can match.

Given this somewhat dismal picture of the current state of classroom instruction, both computer-free and computer-based, what hope can we have for significant educational improvement via the technological revolution? Surprisingly, perhaps, quite a lot. Two factors currently present in the computers-in-education movement have potential for promoting fundamental school change. The first is the fact that computers have entered the schools in a big way, both in the actual number of people and schools affected and in the tremendous interest in the technology itself. The second factor is that the computer's <u>potential</u> for making possible new modes of effective instruction and learning is great. But, unless schools become receptive to--indeed, bring about--fundamental changes in the way they do their work, this educational potential has little hope of realization.

Let us look a little more closely at why these two factors are conducive to school improvement. The first--the big deal surrounding the widespread adoption of computers--indicates that the prevailing view among both school practitioners and the general public is that the computer represents both a substantial educational challenge and considerable educational promise. Why does this seem to make change likely? Certainly, conventional wisdom would lead us to assume that larger changes are more difficult and more easily resisted than smaller ones. However, there is evidence to the contrary. For example, the Rand Corporations's study of factors affecting the implementation of federal programs supporting educational changes in the 70's found that the amount and complexity of change required of teachers in their classroom procedures was positively related to the likelihood of change taking place. The dimensions of these large scale projects that resulted in more overall change included changed classroom organization, curricular revision, and considerable extra effort required by teachers (Berman and McLaughlin, 1977). Clearly the infusion of computers into instruction in existing school subjects involves all three dimensions. The physical presence of the hardware itself requires some organizational rearrangement; the curriculum is certainly revised, if only in mode of presentation; and finally becoming not only computer literate, but a computer teacher, requires considerable effort beyond the usual daily work teachers. What can happen, given the right context--a matter that we will return to shortly--is that as these "adjustments" are made, more profound changes can occur. Once we are in the midst of physical and organizational rearrangements, other areas of the curriculum come under scrutiny.

We know that physical arrangement of the classroom has substantial influence on its social organization. So, while we are moving the furniture, we might reflect on what types of configurations support the kinds of human interactions that are most conducive to academic learning and to the social and personal development of students. For example, the power of cooperative learning groups might be explored with small groups of students working with a single computer terminal. Further, as we alter the mode of instruction from textbook/workbook to software we might consider whether the content we now teach is what we really want students to encounter. We might even question whether we want to continue to view learning as the relatively passive acquisition of knowledge created by others. The "big deal" surrounding computers in schools, then, gives some hope for significant educational change.

Second, while as yet not used much, the technological capabilities of computer hardware and software bring some new effective teaching and learning modes within reach. Programs are beginning to appear which encourage the use of higher-level learning skills instead of just testing recall. These "new-breed" programs are called simulations, and, as the name implies, they try to simulate either realistic problem-solving situations, or encourage the manipulation of objects in a highly controlled, self-contained mini-world. These programs offer students classroom experiences which never were available before computers. A graphic journey through the human body or the workings of an internal combustion engine can be controlled based on a student's response in given situations. The learning potential of these new modes is both exciting and challenging to educators (see for example, Dwyer, 1980).

One of the most common forms of simulation is the adventure format, where students take on the role of an explorer or fictional character and plan strategies to solve problems which are thrust upon the character. What a wonderful opportunity for groups of students to interact and cooperate in problem-solving situations. But how are adventures currently being used? If they are used outside of normal classroom hours or as a reward for "faster" students who finish their normal assignments, then the students most lacking in these problem-solving skills are the students least likely to use these programs.

The most well-known of the "mini-world" simulations is the LOGO language, developed at MIT over the last two decades. In the "world" of turtle geometry, a turtle pointer moves about the screen drawing lines depending on instructions provided by the computer user. One of the important characteristics of this is that beginning from the earliest simple instructions to the turtle the user gradually develops a full-fledged computer language which can be used for many non-turtle applications. Procedures--program instructions developed by students (i.e., the steps the turtle must go through to draw a square)--can be given a name and nested in the computer to build other, more complex procedures). The potential is great for LOGO, (Papert, 1980) but today it is used almost exclusively by individual students for creating geometric drawings. Little of the rich verbal interaction of which the language is capable is currently being exploited. Probably because LOGO goes beyond familiar classroom practice, teachers seem to limit LOGO's use. Potentially revolutionary, LOGO's full potential has yet to be realized in today's school.

The complex nesting structures of LOGO can be used to significantly enhance traditional CAI programs as well. Take a traditional tutorial program and add the ability to evaluate progress and nest sub-programs, and you get a sophisticated system that can address individual differences. Such a program can identify a student's difficulties and branch to sub-programs which address particular areas needing remediation. Programs of this scope generally are available only on large mainframe or mini-computers, as the memory needed to store all of the sub-programs is greater than that currently available on individual micro-computers. Yet, when clusters of micros are networked to a hard disk drive there is access to hundreds of times more memory than is contained within the single computer itself. Once hard disk drives make their way into school computer use, at least the technological barrier to rich multi-level courseware will have been overcome.

Other technological innovations to use the basic microcomputer include speech capabilities and light pen devices which allow simple and quick interactions by just touching the screen with the pen. In addition, video disk interfacing holds tremendous promise for classroom utilization, and as the component prices lower and the sophistication and access speeds increase, we can expect to see more multi-modal instruction which will address many more learning variables that can now be addressed.

But, of course, just because an innovation is perceived as large in scope and has the inherent capacity for significant educational change, it does not follow that change automatically occurs. We have a long history of just such innovations that resulted in very little alteration of either the content or mode of schooling. The "new math" of the early 1960's is probably the most frequently cited "failed" innovation with so much educational promise. But there are many others: open classrooms, non-graded schools, team teaching, PSCS Physics. Anyone in schools during the last twenty years could easily expand this list. Probably closest to the computer-based instruction is, of course, educational television, a widely heralded technological innovation that became a costly and embarrassing schooling flop.

What happened? These educational innovations did not suffer from the lack of good ideas, nor from the absence of considerable enthusiasm about them. What was missing was an appropriate perspective on how change happens in schools and the specific implementation strategies that flow from it.

The introduction of school innovations for the last twenty years were guided nearly exclusively by the Research, Development and Diffusion model. The RD&D process usually begins with the development of a sound educational innovation that meets the needs of those schools. However, it is policy makers who study it, determine its effectiveness, and mandate its implementation. But what of the people, primarily teachers and students, who are in schools, and the objects of the proposed change? The innovation loses its power because it gets disseminated by "experts." Usually, it is presented as <u>the</u> answer to particular problems, an answer that consists of a list of specific teacher behaviors and classroom or school characteristics. The "expert" sets about to have them understand, or at least adopt the changes, with little, if any, input from teachers. Input, when it is solicited is usually gathered <u>after</u> the genuinely important issues have been settled.

Innovation brought to schools in this way comes from available research and development outside the school. Different marketing strategies are used to "sell" innovations to individual teachers. Schools, then, become passive targets for particular innovations. A single aspect of the school or classroom comes under close scrutiny as the focus of innovation. Thus innovation tends to be applied to isolated elements, rather than integrated into the whole of schooling. When attention to the innovation subsides, as it usually does before long, attention to that part of schooling diminishes also. Ways of getting teachers to change, rather than changing the conduct of schooling itself, are the focus. The RD&D perspective is lacking because both its focus on changing individual teachers and consideration of only a small part of the school's functioning do not contend with the realities of how schools resist or effect change (Goodlad, 1975; Sarason, 1982). Most of the reforms of the 60's and 70's assumed an RD&D perspective. Consequently, these innovations did not effectively penetrate the classroom.

Following the argument laid out in Goodlad (1975), an alternative approach to school change proceeds from a culturally responsive perspective. The differences between this view and RD&D are several and important. First, in the culturally responsive view the purpose of change activities is to create a self-renewing school--a school staff that works together to examine the conditions of the school, identifies problems, and develops alternatives based on their own experience and on research in the field. The self-renewing school may use ideas from the outside, but the intention is not to make the school a better target for innovations developed outside of the school.

Second, the primary focus is on the dispositions of teachers and others in the school toward the processes and concepts of change rather than on changing specific structures or teacher behaviors. Having the school staff critically examine the assumptions they hold about how schooling should and can best proceed, together with information about what actually happens, is a necessary condition for bringing about solutions that respond to the problems schools face. But, since schools are vulnerable to social and political pressures from both inside and out, the culturally responsive perspective recognizes they need a great deal of support and encouragement if they are to attempt anything beyond day-to-day survival (Heckman, Oakes, and Sirotnik, 1983).

The first crucial factor, then, when schools themselves begin to implement "innovations" such as computer-based or assisted instruction is the ownership of the innovation by the staff, including an active role in the development or adaptation of the substance of the innovation as well as in the plans for implementation. Second, a great deal of support must be available--support that is viewed as long term and non-judgemental. This support can be viewed as a scaffolding built around a school to both support it during the change process and to protect it while it is particularly vulnerable.

How then does this culturally responsive view translate into ways schools can successfully integrate computer courseware into their curriculum? First those at the school must be central in designing or adapting the "substance" of the courseware itself, making it appropriate to the needs of the school and its students. In this way, not only is an appropriate innovation developed, but it is one "owned" by those who will use it. Second, these efforts must be supported at the school with time and resources for the development activity and with a sense of the importance of the project. In addition, support in the form of ideas or resources from outside the school can help raise the substance and the process of the innovation beyond conventional wisdom and common sense assumptions that develop when a school staff is isolated from theory and research.

What are some of the practical considerations which will arise when teachers examine the curriculum in light of the current research and their own experience? If lucky, the determination of those areas most in need of change will include areas which may be seen as "easy" to change. Here, "easy" corresponds with local control, that is, the internal programs and processes controlled at the school level such as bell schedules, room environments, and student tracking traditions. If the staff, teachers, administrators, and parents work together, such logistical features can be altered as part of an overall implementation strategy.

But the more substantive type of change is also critical and probably more difficult to accomplish. A change in the substance of the curriculum usually collides with rather rigid counter-expectations at many levels of the educational and social community. The school district office, for example, will have to contend with its schools diverging as they meet broad district goals in ways consistent with the unique needs of and talents at each school. Teachers may clamor for transfers from or to the affected school, and parents may demand that their child attend or be transferred from the school. Further resistance to school-directed change will surely come from state departments of education, which have developed frameworks for specific courses and testing mechanisms for school evaluation. And, the external pressure of universities which have begun to dictate the content of high school courses as well as the grading procedures to be used will have to be dealt with. These are mighty hurdles to surmount unless enough support is provided along the way.

There are ways around these problems, and they depend on every component of the educational community doing its part to at least lessen the blows, if not actively encourage change in the schools. Communication between and within each component is critical. School administrators must first create a forum for teachers to meet and grow as professional educators. Teachers must be encouraged to question current practices without fear of being labeled as trouble-makers. Time must be provided for curricular questions to be addressed and those involved in educational theory and research (those from university schools of education, for example) must be called in, not as "change merchants," but as facilitators and experienced partners in change.

The school district must, in addition to scheduling release time for teachers to develop curriculum (both consistent with broad district goals and reflective of needs and talents at the school site) ascribe a sense of importance to this task, and encourage the trying-out of new ideas in the classroom. And in universities, the training ground for many school practitioners, schools of education and liberal arts must work together. Traditional divisions must be bridged with communications such that one does not encourage curricular change while the other resists it.

But even with all of these elements in place, will teachers be able to create intelligent, exciting, and sound computerized educational programs? Although many teachers claim that they have written such materials, there are very few programs that can function both as integral parts of the curriculum and as thought-provoking, active learning tools. A teacher generally will not have the programming skill needed to make a program exciting, while a programmer, although well versed at graphics, animation, and sound techniques, will usually not have the background and experience to develop programs that are both educationally sound and uniquely fitted to a community or school's classrooms. Traditional authoring programs may include some benefits of both professional programmers and educators, but nothing exciting or important has yet been produced by such structured programs.

Much earlier we asked a conventional question about how computerized education could resemble or replace human-to-human teaching and learning. We have eroded the significance of that question with reference to the work of Goodlad and others which suggests that much of what is known to be the strength of that "human-to-human" interaction doesn't occur very often in classrooms anyway. An interim conclusion we can draw is that if education could approximate current human-to-human teaching, and in someways it can, then not much would be benefitted or lost because truly significant lasting changes rarely take place in schools. We then introduced what we and others have found to be true about how change can take place in schools, and we are prepared to offer a new question: How can the spirit of innovation and enthusiasm for change associated with the computer technology bandwagon complement long-needed changes within the educational community? Stated practically, "How can we keep computers in the classroom as one vital component of meaningful school improvement and out of the closet with the learning kits, teaching machines, video equipment and other flotsam of failed innovations?"

At UCLA's Laboratory in School and Community Education a project now underway attempts to confront the problems and possibilities of school change with computerized education. Using the culturally responsive view of school change as a model, the central purpose of the project is to investigate whether a collaboration of public school teachers and university-based researchers can result in the conceptualization, development and implementation of exemplary computer curricula in basic subjects at local school sites.

The work of the collaborative team did not begin with the development of courseware, however. An intensive examination of current curricular beliefs and practices, a survey of research and theory in language arts and mathematics education, and the conceptualization of what curriculum would be "ideal" for students preceded any computer-specific work. The team did extensive reading in the area of computer applications in education and surveyed the extant curriculum software as well. In other words the essential elements of the culturally responsive perspective on change constitute the heart of the project: the examination of current school practice, the critical scrutiny of assumptions about what teaching and learning should be, and the local development of educational alternatives.

During this initial phase of the project, careful steps were taken to insure that the necessary supports for project schools were secured. Both district and site level administrators were a part of the initial conceptualization of the project itself and made a substantial commitment to the importance of the project to their schools. In addition, certain material resources have been designated for project use. The school districts agreed to provide both time and resources (principally equipment) for the project teachers. The school principals arranged working space at each school and facilitate release days for meetings of the entire team. These internal supports help the teachers view their involvement in the project as meaningful and the results of their work as important to their schools.

Additionally, the Laboratory at UCLA provides additional support and resources. Five weeks of intensive planning, reading, thinking and discussion for the entire team were provided at the Laboratory before the project year began. The teachers are considered part of the Laboratory staff as well as members of their school faculties. Summer salaries and part of their teaching salaries are paid from project funds. Importantly, the involvement of the university begins to provide the necessary scaffolding of support for schools where teachers are attempting to try new ways to confront educational problems. The researchers, too, provide access to ideas that go far beyond conventional wisdom and common sense assumptions about teaching and learning in language arts and mathematics and about the use of computers in schools.

During the development phase, work of the project takes place at both the university and at the school sites. The first activity of this phase was the teachers' translation of the "ideal" curricula--conceptualized during the summer--into learning experiences that could be "tried out" in their classroom. Following these trials the series of lessons were examined as to how computers might enhance them and, of course, the learning of students. By developing learning activities from a computer-free perspective, the project is driven by curricular ideas rather than by the limitations of computer technology. Throughout the project we are careful to address curricular issues first, and then try to adapt the technology of computers to them.

An experimental authoring system which allows considerable curricular flexibility while providing the best of the microcomputer technology--color graphics, animation, and sound--is being used by the teachers to adapt their curricula to computer courseware. The system is sophisticated enough that students can be lead through the curriculum in a manner which allows for their varied learning styles and backgrounds. Branching and nesting of programs enable all students to experience a common curriculum without the "holding back" or "hopelessly lost" syndromes teachers are afraid of in heterogeneous classrooms. Exciting lessons and graphics are relatively easy to create, without the need for professional programmers.

Later in the project the team will "try-out" these computer-based learning activities with students and comparisons to determine the relative strengths and weaknesses of the computer and non-computer learning activities will be drawn. As the project continues team members hope to develop their knowledge of how computers can assist in carrying out the "ideal" curriculum and with increasing skill create computerized learning experiences toward this end.

By addressing the process of change in a way that will encourage teachers to own this innovation and providing them considerable support, the project seeks to allow teachers to explore and consider changing areas of the curriculum that have resisted change attempts of the past several decades. We hope that we can, through this process of change, and aided by the awesome potential of the computer, create a self-renewing environment in our project schools that will make future change a much easier and non-threatening task.

References

Becker, H. J. School uses of microcomputers: Report no. 1 from a national survey. The Journal of Computers in Mathematics and Science Teaching, 1983, 3, 29-33.

Berman, P. & McLaughlin, M.W. Federal Programs supporting educational change: The findings in review (Vol. IV). Santa Monica, CA: The Rand Corporation, 1975.

Dwyer, T. The significance of solo mode computing for curriculum design.

Goodlad, J. I. The Dynamics of Educational Change. New York: McGraw-Hill, 1975.

Goodlad, J.I. A Place Called School. New York: McGraw-Hill, 1983.

Heckman, P.E., Oakes, J., Sirotnik, K.A. Expanding the concepts of renewal and change. Educational Leadership, 1983, 40, 26-33.

Hoetker, J. & Ahlbrand, W.P. The persistence of the recitation. American Educational Research Journal, 1969, 6, 145-167.

National Commisssion on Excellence in Education. A Nation at Risk: The imperative for educational reform. Washington, D.C.: United States Department of Education, 1983.

Papert, S. Mindstorms. New York: Basic Books, 1980.

Sarason, S.B. The culture of the school and the problem of change. Boston: Allyn and Bacon, Inc., 1982 (revisied edition).

Sirotnik, K.A. What you see is what you get: consistency, persistency, and mediocrity in classrooms. Harvard Educational Review, 1983, 53, 16-31.

Stevens, R. The question as a measure of efficiency in instruction: A critical study of classroom practice. New York: Teachers College, Columbia University, Contributions to Education No. 48, 1912.

THE SCHOOL DISTRICT ROLE IN INTRODUCING MICROCOMPUTERS: A CONTINGENCY APPROACH

by

Richard C. Williams, Adrianne Bank and Carol Thomas

Introduction

In the field of education one can find computer optimists and computer pessimists. Computer optimists visualize schools of the future as part of large scale networks allowing students and teachers access to information of a quality and quantity never before considered possible. They see computers rectifying the resource disadvantages of small schools, meeting the needs of minority populations, encouraging problem solving, creativity, and individualized instruction. Computer pessimists, on the other hand, see reading and writing devalued as more time is spent with computers and less time with books, greater personal isolation as learning increasingly occurs through interaction with machines rather than with people, and a widening gap between the rich who have computer access and the poor who do not (Coburn, et al., 1982).

But whether an optimist or a pessimist on this issue, a school or district administrator must be a present day computer realist. According to Market Data Retrieval, figures from a 1982 telephone survey of all U.S. districts show that over 24,000 public schools now use microcomputers in instruction. This is a 60 percent increase from the previous year with the fastest growth rate occurring in elementary schools. And, of the 15,314 districts in the U.S., 9,245, or 60.4 percent, had microcomputers in 1982 as compared with 6,441 a year earlier. The rate of growth was highest in the smallest districts. Fifty-two percent of these small districts -- under-1,200 students -- had microcomputers while almost three-quarters of the districts with over-10,000 students had them (Market Data Retrieval, October, 1982).

The rate of growth in schools and school districts' acquisition of microcomputers is phenomenal and is expected to continue. But the current statistics on the availability of hardware may be misleading. The National Center for Educational Statistics reports that computers were used by an estimated 4.7 million students during the 1981-82 school year which averages to only 9 hours a year of computer access for each student. Differences by grade were noted in the amount and type of use. While high schools cite computer science as their major use in instruction, junior highs use terminals for remedial instruction, enrichment and computer literacy. In elementary schools, terminals are used mainly for enrichment, remediation and basic skills instruction (National Center for Educational Statistics, 1982, p. 2).

There is great variation, then, in the availability of personal computers in schools and in the uses to which they are put. There are also vast differences in the role which district offices play in introducing computers into their educational setting.

Some districts, especially large districts or those with strong administrative staffs, have employed a centralized strategy to introducing computers. Here, the district office directs the process of selecting, funding, and placing microcomputers in schools, usually with some input from principal or teacher participation. The advantages of such a centralized approach include: 1) the development of conveniently located and deployable expertise for training and troubleshooting in many schools; 2) the increased capacity to coordinate hardware, software, and training; and 3) the added clout the district has in negotiating with vendors on price, service contracts, and software when they purchase in bulk. Disadvantages of the centralized approach include: 1) diminished teacher "ownership" of and enthusiasm for both the hardware and software, 2) less flexibility in accommodating specific classroom needs for particular kinds of hardware or software, and 3) lack of administrative knowledge at the central office level.

Other districts have, either inadvertently or intentionally, used a grassroots strategy to introducing computers. Often, in these cases, computer buffs among the teachers learn as much as they can, find their own funding or apply to the district for funding in order to buy and use computers in their own classrooms in their own ways. District administrators reason that enthusiasm will spread to other teachers who at some point will come together to form a school-wide plan for themselves.

Advantages of this strategy include its low cost to the district in terms of educating personnel and grappling with individual schools' problems and the natural spread of the innovation because of enthusiasm and individual initiative. Disadvantages, however, may be serious: much money may be spent on hardware and software while only a few children will learn particular skills; these skills either may not be picked up in subsequent grade levels or subject areas or may be unnecessarily repeated.

Between the extremes of the centralized and the grassroots strategies are many combinations. Each district where the computer issue has arisen -- and we should note that close to 40 percent of the districts have not yet grappled with the situation -- seems to muddle through, formulating its own responses in reaction to various kinds of pressures.

The Growing Importance of the District Role

Our argument here is that the district central office, along with its school board, must play a crucial role in introducing microcomputers into its schools. Mistakes are becoming increasingly costly. Some districts have rushed out to buy microcomputer systems and found, unhappily, that the system they bought will not meet their future needs and that even their present instructional programs are not well served by their system (Thomas & McClain, 1981).

It is clearly unfair to students not to solve issues of equality of computer access. Another survey conducted by Market Data Retrieval (1982) found that school microcomputer use is associated with wealth of the district -- 80 percent of the nation's 2,000 largest, richest high schools used microcomputers, while only 40 percent of smaller, poorer high schools had them (Lipkin, 1983). Use is also associated with gender. A survey of 10 New Jersey high schools offering computer courses revealed a consistent dominance of male enrollment of slightly more than 60 percent. Studies of California schools report a similar trend (Bakon, et al., 1983). In addition to computer access, the issue of equity is raised by what schools use computers for. Computers in suburban schools are often used to teach programming and computer awareness. Computers in less affluent inner-city or rural schools are more likely to be for drill and practice and remediation (Field & Kurtz, 1982; Lipkin, 1983). The desirability of encouraging a coherent computer literacy scope and sequence, analogous to that in reading, math and language arts for girls as well as boys, in poor as well as wealthy schools, is daily becoming more evident. There are, in short, many issues surrounding computer acquisition, deployment, and use that are too large and too complex for individual schools to resolve each in their own manner. In the current context of educational computer use, effective districts are essential for effective school utilization.

A No-Plan Approach to District Computer Involvement

As noted above, many districts have responded reactively to the rapidly expanding availability of relatively inexpensive computers and programs that can be used for managerial and instructional purposes. Whether centralized or grassroots in character, their approach might well be labeled "non-planned."

There are a number of understandable reasons for the prevalence of this approach. School districts, like many individuals and other organizations in the public and private sectors, are unsure about how to assess the potential value of an "exploding" technology. They are bombarded by the marketing strategy of vendors. For example, many computer companies provide free or low-cost introductory offers to school districts in hardware, software or staff training in order to get districts to make a long-term commitment to the vendor's brand. School districts operating on meager financial resources find it difficult to refuse the hook hidden in this sudden technological largesse. Many purchase before they plan.

Another reason for the "no-plan" phenomenon is avoidance: the level of uncertainty and ambiguity is so high among central office staff that they don't know where to begin. The hardware and software is constantly changing; is unfamiliar to many who would potentially benefit from its availability; threatens some who think they don't want to or can't learn about it. A further psychological complication is created by students who seem to know far more about and have far greater aptitude and appetite for this new technology than do administrators or teachers.

A Linear Approach to District Computer Involvement

To some administrators, the logical antidote to "no-plan" is to begin a linear planning process following a series of sequential steps that would include: carefully defining the district's objectives as regards computer use; determining those steps that would have to be taken by various district components, e.g., teachers, district administrators, principals, in order to accomplish each objective; establishing time lines and sequences to be followed; determining ways to evaluate whether specific objectives had been achieved; applying corrective actions in instances where objectives had not been met.

Linear planning can be an effective tool to help organizations achieve specific goals when there is a common knowledge base, where lines of authority are clearly defined, and where there are the resources to carry out the implementation sequence; we doubt, however, that linear planning is an appropriate tool for the rapidly changing computer situation. School districts lack sufficient knowledge about or control over important factors that must be accurately estimated in order for a linear process is to dictate decision making. For example, school districts are often subjected to shifting forces outside of their organizational boundaries over which they have little control, e.g., political support in the community, changing population, externally mandated strategies in key administrative and instructional areas, and uncertain financial resources. Given these conditions, combined with rapidly expanding computer technology, we think it non-productive to try to determine exact goals and the means to accomplish them. By the time a comprehensive objectives-based plan is devised, it is likely that conditions will have changed so as to make the plan obsolete.

Under such conditions of uncertainty and change, we reject both "no planning" and "linear planning." We suggest instead the use of an intermediate scheme which we will refer to as contingency planning. This approach suggests that districts' planning be ongoing, incremental, adaptive and self-correcting.

A Contingency Approach to District Computer Involvement.

While traditional planning is based on events that have a high probability of occurring, contingency planning takes into consideration other likely conditions, which, if they actually occured, could create serious difficulties for a school district. A contingency approach prepares administrators to take specific actions when an event or condition not anticipated in the formal planning process actually does take place. Such forethought reduces uncertainty and delay, and makes responding to the unpredictable a reasonable part of daily life.

A contingency approach identifies issues of concern (e.g., "what if" questions) and estimates the probability of their occurrence (Steiner, 1979). Both the degree of criticality and the degree of probability of occurrence must be considered. Alternative strategies to deal with the possible occurrence of likely events are identified and considered in terms of the anticipated nature of the events and the district's capabilities and constraints in dealing with them. The result of such strategizing may be a district staff's decision to take some advance "damage control" actions as well as to identify actions to be followed at the time of the events.

A contingency approach may describe "trigger points" or those warning signs which would signal the imminence of the events for which contingency plans have been developed (Steiner, 1979). In some cases, the trigger point might be the event itself, but in other cases the point at which some action should be taken predates the event.

For example, using a contingency approach, district administrators themselves should begin or continue to become knowledgeable about a wide range of computer-related topics from technology to staff needs, attitudes, and purposes. At the same time, the district administrators should become aware of present uses of computers. They should start to imagine alternative scenarios for accommodating the district ways of doing things to the demands of the new technology.

Prepared with data and with scenarios, the district administrators should identify the optimal dates by which they must make critical decisions regarding what computers to buy, when they should be bought, who should use them, and who should have them. In other words, many of the district's future plans and actions will be contingent upon the unknown opportunities that may emerge at some indeterminate point in the future. It is a complex task to decide in advance not only when to act, but how.

Components of a Contingency Approach

Gearing up for the near and intermediate decisions about computer use in the district should happen in three ways: 1) conducting a situation audit of the external and internal environments; 2) generating within-district support; 3) formulating district-wide policies; and 4) developing an ongoing operational plan.

1. Conducting a Situation Audit

The term situation audit refers to a systematic analysis of data, past, present, and future (Steiner, 1979). Such an audit provides the base for planning computer purchases, deployment, and use. The potential range of topics to be covered in a situation audit is wide and includes anything of importance in the internal and external environments. A major objective of the situation audit is to identify and analyze the key trends, forces, and phenomena that have a potential effect on the formulation and implementation of a framework for district computer use. The situation audit also provides a forum within the district for discussing and debating divergent views about what are the relevant issues likely to impact policy and operations. We discuss the situation audit in terms of an internal inventory and an external resources listing.

An internal inventory. In order to develop an effective district framework, administrators need to know what is already occurring in the community, schools, and homes of students enrolled in the district.

Through surveys and telephone interviews, baseline information can be collected regarding what equipment is currently available, how much it is used, what resources and skills there are at present in the district.

Detailed information is needed on schools' current inventory of types of hardware, maintenance problems, servicing costs, support from vendors, the extent to which existing hardware is compatible and expandable. District administrators should know what software has been purchased, where it is stored, how much it has been used. In addition, the district will need to know who, at each school, is overseeing the use of the computers; how computers are being used, and for what percentage of time by which students. In California, one district, inundated with a variety of microcomputers, conducted a survey to determine what equipment existed in their schools. They found that during the past few years each secondary school department had been acquiring its own equipment to meet specific needs. This piecemeal acquisition was now creating problems since schools had bought different brands (Stremple, 1983). The survey indicated a need for district level policies about the purchase of additional equipment.

Staff in the district office should be surveyed to determine who has skills for operating what equipment and software, who can program in various computer languages, who can be a trainer of trainers, demonstration teacher, or software evaluator. Parents of students enrolled in the district should be surveyed to determine if a computer is in the home, what type, and if it is used by the student. Finally, an attitude inventory should be used to discover teachers' and students' attitudes towards computers.

District staff should also determine what information is already being systematically collected by the district that might be helpful in making decisions about computer use (e.g., existing instructional programs, demographic profile, student achievement data, financial transactions, etc.) and include this information in the data base.

An external resources listing. In addition to internal resources, there are many groups and agencies external to the school district that might provide assistance to district staff contemplating computer use. Electronic Learning magazine (1982) conducted a survey that identified 38 statewide educator-user groups in 33 states, all of which have the general aim of promoting the effective use of computers in the classroom. In those states where no statewide groups were identified, often a special unit within the state department of education was filling the role. These groups varied in the services they offered, among them: cooperative funding, newsletter publication, conference organization, resource center, inservice training, software library, and software evaluation. A few of these user groups have national memberships. For example, school teachers in the Santa Clara County area of California formed the Computer-Using Educators (CUE) group which has a membership of over 700 people in 19 states (Unseem, 1981). Minnesota Educational Computing Consortium (MECC) provides services to Minnesota schools and schools in adjoining states.

Corporations and industry leaders also provide support to school districts. Hewlett-Packard in California has fostered industry-education ties by having a number of full time employees who devote time to improving the company's contact with public schools. A committee of top executives examines ways the firm and industry can provide more support for public education. They have loaned personnel and given equipment to schools (Unseem, 1981). A partnership exists between the Washington, D.C. schools and Control Data Corporation. Their partnership calls for the firm to donate $118,000 worth of terminals and software and an equivalent amount worth of training and administration to the school district. The school system will be matching that contribution (Education Daily, 1982).

Some organizations provide services to districts in specific areas of computer use such as software evaluation (e.g., MicroSIFT in Oregon); information exchange (e.g., Association for Educational Data Systems); data bases (e.g., Resources in Computer Education [RICE]); and newsletters and magazines (e.g., The Computing Teacher, School Microware Directory, Software Review).

In its survey of external resources, the district should become knowledgeable about the talents, skills, and attitudes of people living within its attendance area such as merchants, industry specialists, and consultants.

2. Generating Support Within District

This is a top priority. The biggest problem technology enthusiasts had a few years ago was convincing educators that there was a need for computers in our schools; today, in many districts, that is no longer such an obstacle (Oliver, 1983). However, commitment from groups such as board members, parents, administrators, teachers, industry and community leaders, and other educational resource agencies is necessary to build a district-wide policy consensus. A network of interested persons can be a continuing support system for services, equipment, or funding.

Successful strategies for generating support for a computer policy vary from district to district. Hands-on experience helps. In some districts, having computers available for home experimentation by teachers and principals has been effective. Establishing demonstration sites so that board members, principals, and teachers can have the opportunity to see computers in operation and experiment with them has worked in other districts (Swalm, 1983). School districts have loaned school computers to parents over weekends and holidays. A large school district in Texas initiated a computer project that offered low-income parents and children 12 hours of instruction, after which parents could check out computers for home use (Sturdivant, 1983). Other school districts have organized computer fairs, computer clubs, and computer competitions to increase public and student interest (Fisher, 1983).

An enthusiastic "idea champion" in each school may persuade other teachers to consider approaching the computer supporters in the district. One Texas district developed a new job role called "teacher technologist" for each school (Sturdivant, 1983). Resource centers and use groups have also been formed to share information between schools (Useem, 1981; Stremple, 1983; West, 1983).

Idea champions in the district office are also critical to the success of any computer use plan. In some districts, administrators have created formal units to address issues and allocate resources. For example, the Houston Independent School District has a new division called the Department of Educational Technology that is responsible for implementing a district-wide plan for computer use (Oliver, 1983).

3. Formulating a District-wide Policy Framework.

Essential to the success of a district computer program is the formulation of a framework to guide the subsequent development of an operational plan. Evolving such a framework allows the district to examine all aspects of computer use and then to decide on the best applications for students in the district.

With district administrative support, an inter-school committee can be organized and charged with the responsibility for developing the district's policy framework and program goals. If the district wants computers to be used by all teachers, the committee should not be dominated by any one subject area (Swalm, 1983). The committee should include representatives from interested groups while remaining small enough to constitute an effective working group. One large district with a committee of 25 members took two days to agree on only four goals related to computer use, while another district committee, with seven members, wrote the entire plan in one day (Fisher, 1983).

The goals of the district computer program will facilitate the definition of school-level objectives. Goals for a district's computer use might include: to develop computer literacy for each student by the end of grade 8; to provide equity regarding computer use; to provide an inservice program for teachers, partents, and administrators; to use computer-assisted instruction for remediation in basic skills; and to evaluate the district's involvement in the use of computers.

In deciding upon district policies the committee needs to list the big picture issues it will discuss. In doing so, it should decide whether the central office or the schools will make the decisions on those issues and whether the decisions should be made immediately or put off until sometime in the the future. A brief summary of some of the issues with which a district must eventually cope is included in Table 1. We have grouped these issues into categories: hardware acquisition/fiscal issues, software issues, management issues, staff development issues, and instructional issues. The issues in each category have been organized according to major policy questions and operational planning questions.

A second task of the committee is to develop an overall timeline. Districts that have successfully integrated computers into school programs usually construct policy frameworks that spread implementation over several years. West (1983) found that the best way for their district to incorporate computer literacy into the curriculum was to develop a five-year framework setting goals and objectives in instructional and management applications. Fisher (1983) suggests that a long-term framework is more effective than a one-time plan. According to Fisher, having a framework spanning several years signals a continuing commitment by the district to computer use and is visible evidence that teachers can become involved in the planning at several stages of the process. A long-term framework can also aid in reducing the fiscal burden in any one year.

General financial planning should go on concurrently with developing a framework. A common error in financial planning is to think only about the initial direct cost of the computer facility and the hardware. Larer and Moursund (1980) list other aspects that take time and require financial resources:

Table 1

Issues in Need of District Attention

Issues	Policy/Framework Questions	Operational Planning Questions
Hardware Acquisition/ Fiscal Issues	° What criteria/guidelines should be established for hardware acquisition? ° What percent of the computer budget should be allocated for software purchase and maintenance? ° Should a single computer system be used for both instructional and administrative purposes? ° What percent of the computer budget should be allocated for software purchase and maintenance? ° What resources are available for personnel costs associated with hardware use? ° What inservice training budget allocations should be made? ° What strategies should be used by educators in dealing with computer vendors?	° What successes/failures have been experienced by other districts with specific hardware? ° What is equipment's reliability? ° What maintenance warranties and assistance will vendors provide in installing and servicing the equipment? ° What peripherals are available for specific hardware and provided by the vendor? ° What expansion options exist? ° What training will the vendor provide in the operation and programming of the hardware? ° What size machines and/or memory are required to run the programs needed and achieve computer use objectives? ° What software is available and at what cost in relation to the characteristics of hardware? ° What are the estimated costs for hardware, software, maintenance, facility preparation, and staffing needs for each application? ° What strategies should be used for financing computer acquisition?

Issues	Policy/Framework Questions	Operational Planning Questions
Software Issues	° Under what conditions should districts undertake software development? ° Should the district operate a software library? ° What is district policy relating to copyright issues for purchased and teacher-developed programs? ° How and by whom should software be located evaluated, and acquired?	° How can results of software evaluation be disseminated? ° Do the software cassettes or discs include documentation? ° Is the software program educationally sound? ° How can computer software be integrated with other instructional activities?
Management Issues	° What role will other educational service agencies and groups have in the district framework and plan? ° How will the district judge if their computer implementation program is successful? ° How should resources be allocated to ensure equal educational access and use of computers? ° What security precautions should be taken? ° What phasing-in strategy should be implemented for the district's computer plan?	° What implementation strategy and timelines are needed for elementary and secondary levels of the district? ° Should schools have centralized placement or individual classroom/department placement of computers? ° What strategies can districts use to encourage female students in computer use?
Staff Development Issues	° What do teachers, principals, and other district staff need to know to use computers? ° What teacher certification requirements should be established, if any? ° Who should conduct and evaluate the computer training and what type of follow-up assistance will be provided?	° Will the district develop staff to be local computer resource persons? ° What computer training, both preservice and inservice, should be required for teachers and administrators? ° What strategies should be used to allocate time for staff training and hands-on computer experience?

Issues	Policy/Framework Questions	Operational Planning Questions
Instructional Issues	° What roles will computers have in the school, e.g., computer-assisted instruction, computer literacy, computer programming? ° Should all students meet minimum computer competency requirements? ° How will the instructional role of teachers change with increased computer use?	° What kind of social problems are being introduced into schools along with computers? ° How can the district ensure equity in computer use, especially higher level and creative uses? ° What are reasonable rules and guidelines for student computer use? ° Is there a specific need for a "computer literacy" curricula? ° What are appropriate educational goals and curriculum materials for computer literacy? ° How can the teacher overcome the constraints of using individually-oriented computers in the context of a group-based instructional organization?

- needs assessment and general planning;
- writing of specifications, dealing with vendors, evaluation of bids, and supervision of installation;
- site preparation for the facility;
- acquisition of supplies and supplies maintenance;
- maintenance and repair of computers -- a standard estimate is that for large computers a maintenance contract costs about .75% of total equipment cost per month and for microcomputers, perhaps 2% per month;
- operators and a programming staff for large computer systems;
- teacher training and curricula revisions;
- courseware development, revision, purchase, and distribution.

4. Developing an On-going Organizational Plan

Using the policy framework, either the committee or other school or subject groups should develop specific operational plans. Activities which contribute to the ongoing operational plan include: analyzing curriculum materials in computer literacy for appropriateness, investigating and evaluating software, visiting programs in other school districts, attending conferences and vendor demonstrations, and developing staff development strategies.

Instructional objectives. An ongoing operational plan will state general instructional objectives such as:

- students will have the ability to understand the basic part of a computer;
- students will be able to interact successfully with a variety of programs;
- students will be able to create a BASIC program;
- students will be able to demonstrate an understanding of ethical principles in the use of computerized information systems.

Instructional objectives could be stated in terms of types of students, grade level, and subject areas. The goals of the district computer program will facilitate the definition of school-level objectives and determine at what grade level and in what subject areas each should occur.

For example, in a California school district, under the broad goal of programming, modifying computer programs was an objective for math students in grades 6-8 (Fisher, 1983).

In another school district in California, for example, under the broad goal of use/operation of the computer, objectives were given for three subsets of grade levels: K-5 students would learn how to operate the computer, load programs, and respect copyrights; 6-8 students would focus on appropriate computer use, typing, keyboard, and functions; and 9-12 students would spend time on appropriate programs and vocational use, such as word processing, data bases, network, and telecommunications (Fisher, 1983).

In Cajon Valley Unified School District, also in California, all of the 22 schools in the district were asked to submit a statement of assurances specifying how they would use computers, what their goals and student objectives were, how they would evaluate the program, and who would be responsible for their school's computer program (West, 1983).

Staff development. The ongoing operational plan might also include objectives and strategies for staff training necessary to implement the district computer use framework. A school district in New York State developed the following four inservice objectives: staff will acquire a functional knowledge of computers for educational use, staff will learn how to integrate computers into the learning environment, staff will develop the necessary programming skills to facilitate creation of software suitable for classroom use, and staff will acquire the knowledge necessary to teach principles of computer awareness (Center for Learning Technologies, 1982).

Naiman (1982) proposed the following staff development strategies:

- Individual teachers, already knowledgeable, can train others;
- The school or system can provide inservice courses during or outside of class times or on inservice days;
- The state department of education and regional centers can be encouraged to offer computer training;
- Professional associations offer computer workshops at their meetings;
- The school system can provide release time on a regular basis for teachers to take courses;
- Provide sabbaticals for someone in the district to learn and then share expertise with others;
- Colleges offer semester-long courses or weekend workshops;
- Other public or private organizations, user groups, computer stores, manufacturers, and vendors offer occasional or regular workshops.

When instructional objectives are clear, and inservice needs assessed, the committee can investigate and evaluate software, and finally determine what hardware is required (Swalm, 1983). A contingency approach is better than no-planning or lockstep planning. And we approve of Fisher's (1983) admonition to leave lots of space in whatever plans are developed: "A good plan will provide time for schools and teachers to 'get up speed,' to become informed and trained in computer use so they can make effective decisions; it will also leave room for serendipity and individual differences." (Fisher, 1983, p. 13.)

References

Bakon, C., et al. Computer fear. Educational Leadership, September 1983, 41(1), 27.

Center for Learning Technologies. Computer literacy: An introduction. Albany, NY: The University of the State of New York, The State Education Department, and the Center for Learning Technologies, 1982.

Coburn, P., Kelman, P., Roberts, N., Snyder, T., Watt, D., & Weiner, C. Practical guide to computers in education. Menlo Park, CA: Addison-Wesley Publishing Co., 1982.

Public-private partnership brings computers to D.C.schools. Education Daily, June 22, 1982.

EL's Second Annual Survey of the States: Statewide user groups. Electronic Learning Magazine, Nov./Dec. 1982, 2(3), 62-71.

Field, E., and Kurtz, C. Equity emerges as major issue in schools' use of computers. Education Week, April 21, 1982.

Fisher, G. You need a district wide plan. California School Boards, April/May/June 1983, 42(3-4), 11-13.

Larer, J., and Moursund, D. Acquisition of computer facilities. Working paper from the Association for Computing Machinery's Elementary and Secondary School Subcommittee, Oregon, 1980.

Lipkin, J.P. Equity in computer education. Educational Leadership, September 1983, 42(1) 26.

Update on the school market for microcomputers. Market Data Retrieval (Ketchum Place, Westport, CT), October 1982.

Naiman, A. Microcomputers in education: An introduction. Cambridge, MA: Technical Education Research Center, Inc., and Northwest Regional Exchange, Inc., 1981.

National Center for Educational Statistics. Instructional use of computers in public schools, NCES 82-245, September 7, 1982.

Oliver, P. Technology and education: An administrator's round-table. Electronic Learning, Feb. 1983., 2(5), 63-66.

Steiner, G.A. Strategic planning: What every manager must know. New York: Free Press, 1979.

Stremple, J. Computers come to Fresno. California School Board, April/May/June 1983, 42(3-4).

Sturdivant, P. Can technology save our schools? School, business partnerships may be answer. Electronic Education, March/April 1983, 2(7), 20.

Swalm, J. Don't hide your computer in the closet. Electronic Education, May/June 1983, 2(8), 40-41.

Thomas, D., & McClain, D. Selecting microcomputers for the classroom. In J.L. Thomas (Ed.), Microcomputers in the schools. Phoenix, Arizona: The Oryx Press, 1981.

Unseem, E. Education and high technology industry: The case of Silicon Valley. Boston: Northeastern University, Institute for the Interdisciplinary Study of Education, 1981.

West, C. Cajon Valley develops plan. California School Board, April/May/June 1983, 42(3-4), 38-39.

Some New (and Old) Directions for Computer Courseware

J.D. Fletcher

Center for Advanced Technology in Education

University of Oregon

In 1960 T.F. Gilbert wrote:

> If you don't have a gadget called a "teaching machine", don't get one. Don't buy one; don't borrow one; don't steal one. If you have such a gadget, get rid of it. Don't give it away, for someone else might use it...This is the most practical rule, based on empirical facts from considerable observation. <u>If you begin with a device of any kind, you will try to develop the teaching program to fit that device</u>. (p. 478. The emphasis is Gilbert's.)

This is a point of view with which many of us will sympathize. Educators who have mastered their craft through considerable investment of time and energy in learning how to use the traditional technologies of text, lectures, blackboards, and real-equipment laboratories have every right to be suspicious of new technology that threatens to revolutionize the hard-won techniques now at hand. Even programmers, initiates into the priesthood of computer technology, are occasionally elevated by computers to levels of frustration in which they are willing--and eager--to destroy thousands of dollars worth of equipment with their bare hands. Moreover, Gilbert is undoubtedly correct when he suggests that we may develop teaching programs to fit the technology at hand. Of course we will, and to varying degrees we always have. To suggest that we should not pursue new technologies for this reason may not be so correct.

As Marshall McLuhan (1967) pointed out, every technology, to some extent, carries its own message. To ignore this message is to neglect the strengths of the technology. The technologies now becoming available will not only provide powerful new instructional tactics for presenting context, they will also make some content accessible that heretofore could not be taught in any practical setting. In the development of computer courseware it is possible to discern entirely new "functionalities" in instruction. As is true of most technological efforts, we have begun by trying to enhance the capability of our existing practice. We may end with new capabilities that change the nature of what we do in ways that are completely unanticipated. This could be the essence of the new computer revolution in schools. It is not just that we will have computers everywhere or that we will enhance our capabilities to instruct. We may also change our ideas about what instruction is. Not only will we get better at doing what we do now, but in a fundamental sense we may change what it is we want to do.

New Directions

It may be well to begin with a fable. This fable will already be familiar to some readers. Nevertheless, it seems sufficiently relevant to bear repeating. As the story goes, there once was a government "blue-ribbon" commission of instructional experts assembled to specify the ultimate in instructional technology. After several days of meetings--suitably fueled by long lunches and accomodated by comfortable lodging--the experts came up with the following specifications for the new technology:

1. There should be no exotic power requirements. The technology should use ordinary household current, or be battery powered, solar powered, or require no power at all to operate.
2. It should be light and easily portable. One person should be able to transport it, and at best it would be carried in one hand.
3. There should be no complicated installation or environmental requirements. It should be easy to set up and use, it should operate in moderately extended temperature ranges, and it should be, as the miltary says, "ruggedized."
4. It should provide random access to a large amount of material.
5. It should be capable of displaying graphics, photographics, drawings, color, and high quality, easily read text.
6. It should be inexpensive, costing less than $50 a copy.

The commission report was received with great relief, for, as the perspicacious reader may realize, no research and development money was required to develop the technology. In fact, the technology already existed and had been in place for over five hundred years.

The appropriate technology was, of course, a book.

This is a fable for all of us in the business of applying new technolc o instruction. We must come up with solutions that promise real innovations; in the case of instructional technology, they must be better than books. At the same time, some of our prototypes will be, like the horseless carriage, less efficient than what they are intended to replace.

Books are important beause, among other things, they are able to capture instructional content and make it inexpensively available to an unlimited audience. As Bunderson (1981) pointed out, computer technology is important because, among other things, it makes both the content and the interactions of great instruction inexpensively available to an unlimited audience. This promise has yet to be realized, but it seems almost inevitable. What we need to do is sift through all the prototypal development and find therein those embryonic techniques that promise to be better than books. It turns out that these techniques are neither easy to find nor trivial to develop. I will briefly examine them in the three areas of drill and practice, tutorial dialogue, and simulation.

Drill and Practice

"Drill and practice" is doubtless one of the more regrettable terms in instruction, evoking images of the classroom as a sweat shop and attracting the ire of those who want to use computers to create a rich and friendly learning environment for intellectual exploration and discovery in the classroom. Certainly it is now fashionable to deprecate drill and practice as a computer instruction technique, and it has been so for the last five years. Papert (1980) cites drill and practice as an example of the QWERTY phenomenon. It turns out that because the mechanical keyboards of earlier times were unable to keep up with skilled typists--the keys would jam and otherwise misbehave if they were operated too quickly--typewriter keyboards were originally designed to slow down the key presses of skilled typists. The result was the QWERTY keyboard, named after the topmost row of letters. This keyboard is with us today despite our having removed all the mechanical obstacles to fast operation that results in the QWERTY design in the first place.

Papert's argument is that early applications of computers to instruction necessarily followed drill and practice formats partly because that is what classroom teachers would accept and partly because the computer technology of earlier times could support nothing else. This point of view is not entirely accurate, as can be seen in the design of curricula for the IBM 1500 System in the mid-1960's. The Stanford beginning reading program is a case in point. This curriculum, which was designed roughly in the period 1964-1966 and is described more fully by Fletcher (1979), encouraged children to build matrices using words and spelling patterns, to read and to be read stories (with illustrations), to record and play back messages, and to experiment with linguistic forms and variations

Teacher acceptance was an issue somewhat separate from the content and approach of the curriculum--using computers to teach at all and taking away from classroom time to do it were the central concerns of the teachers. Nonetheless it is notable that when the

Stanford group moved to a less expensive machine configuration for presenting beginning reading instruction, the curriculum became more drill and practice in nature.

In any event, it seems past time to make a few arguments in favor of drill and practice. Is drill and practice an example of Papert's QWERTY phenomenon? The answer seems to be "no", partly because it works--drill and practice is still one of the most successful techniques we have in computer instruction--and partly because there is so much yet to be tried and developed in the drill and practice mode. Even if we assume drill and practice is limited to presentation of discrete items such as math facts or vocabulary items, there are at least three directions for curriculum development in drill and practice. These have to do with performance goals, optimal item selection, and optimal activity selection.

Performance Goals

We may best begin with trajectory theory. Basically this is a way of accounting for the progress, or trajectory, of individual students through a curriculum as a function of the amount of time they spend working in the curriculum. Figure 1 shows, perhaps more clearly, what trajectory theory is getting at. For individual students A, B, and C we try to predict and prescribe their grade placement on standardized paper and pencil tests based on the amount of time they spend on the computer curriculum. The interesting thing about trajectory theory is not just that it works, but that it has worked amazingly well in practice. In two published studies using trajectory theory (Suppes, Fletcher, & Zanotti, 1975 and 1976) the standard error of estimated grade placement was in the range .04 - .06 of a grade placement. In other words, the estimates were off by less than a tenth of a grade placement for 90% of the cases. Again, these estimates were based solely on the amount of time the student spent on the computer and were independent of what was being done in the classroom. If we want to predict and control progress toward measured goals of achievement, trajectory theory may be one of the best techniques we have. It is worth emphasizing that although trajectory theory was developed for drill and practice, it may be applied to any form of instruction where we have closely watched and accurate measures of time on task, as we have in computer instruction.

There are still many questions to be answered about trajectory theory. Can it be applied to all subject matter? Can it be applied to methods of instruction other than drill and practice? Are there significant and important benefits to be gained from using classroom observations of time on task as well as computer time to predict and control progress? The list of questions could be continued. Trajectory theory is not a particularly new technique for computer curriculum, but it remains promising and worthy of further development.

Optimal Item Selection

Basically, an instructionally optimal solution is one that attempts to maximize some outcome, such as scores on an achievement test, subject to some constraints, such as total time on task, session length, and student ability. Optimal solutions are brought to use by control theory which in turn comes from operations research. It is a well known and noted fact that operations researchers tend to attack problems by removing from them everything difficult to quantify or measure and building an imposing mathematical structure on what remains. In the current instances, the imposing mathematical structures remain, but some portion of what is difficult to quantify or measure can be supplied by mathematical models of learning and memory. The wherewithal for applying both these models and control theory to instruction in real time is provided by computers in the context of computer instruction.

The problem of optimal item selection in instruction was stated with mathematical clarity and rigor by Suppes (1964), but can be stated fairly simply in words: given a large number of items to be taught and a fixed time in which to teach them, what subset of items should be presented to an individual student at a given time in order to maximize his or her eventual achievemen. The answer can be supplied by the above-mentioned quantitative models of learning and memory. Figure 2 presents a probability state-transition matrix of an appropriate sort based on General Forgetting Theory (Rumelhart, 1967; Paulson, 1973). This matrix shows what can happen to an item when it is presented to a student. As can be seen from the figure, the model of learning postulated is very simple. If an item is in the learned state, it stays there. If it is in the short-term state, it will either advance to the learned state or stay where it is. If it is in the unlearned state, it will advance to the short-term state or the learned state or remain unlearned. General Forgetting Theory is actually a little more sophisticated than this in that it accounts for probabilities of correct responding separate from the learning status of items and, notably, it postulates what happens to the learning status of an item when it is not presented. An optimal strategy for item selection based on General Forgetting Theory is, like all models of this sort, fairly simple in its view of human learning but fairly complex to implement. It could not be implemented by a book.

Studies by Lorton (1973) for teaching spelling and by Laubsch (1970) for teaching foreign language vocabulary have shown approaches of this sort to be effective. They may even be dramatically effective, far more so than any other method for teaching large numbers of relatively independent items to students, but little work has been done in them since the mid-1970's. It seems to be a thread of research we have let slip through the cracks. There seems to be no real reason to drop it from our list of new directions for computer curriculum. Its promise for exceedingly efficient instruction remains.

Optimal Activity Selection

A few words may also be in order for optimal selection of activity. This problem most clearly emerges in the context of "strands" approaches to curriculum development. The strands approach, which was first described by Suppes (1967), calls for the apportioning of a computer curriculum into various content areas, or strands. For instance, a curriculum in reading comprehension might be divided up into vocabulary, literal comprehension, and interpretive comprehension strands. The problem, then, for a computer curriculum designer is to decide how much time students should spend in each strand or, to state it a little more completely, how to control student progress in each strand so that each student's achievement over all strands is maximized at the end of some specified period of time. If progress in each strand is independent of progress in each of the others and if each of the strands contributes equally to the measure of achievement, then the solution is simple: we just pick the strand in which learning rate is greatest and allocate all the student's time to it. If, however, the situation resembled our reading comprehension example in which progress in one strand is interrelated with progress in the others, the situation is more complex. In reading, after all, a student with a poor vocabulary will not progress very far in literal or interpretive comprehension, yet the achievement measure of success for the curriculum will presumably be more concerned with comprehension than with vocabulary growth. Some sort of optimal mix of vocabulary development and work in comprehension will have to be devised for the student.

An appropriate optimal strategy (based on the Pontryagin maximum principle of control theory) for adjusting progress in interrelated strands was devised by Chant and Atkinson (1973) for the case of two strands. This strategy determines how much time a student should spend each day in each strand, depending on the student's learning rate in each strand and on how much he or she has progressed already in the strand. Extension of the strategy to curriculum environments with three or more strands was left by Chant and Atkinson as an exercise for the reader, but was described by the authors as being "straightforward". It very probably is, but it has not been done, or at least it has not been published. Moreover. there have been no applications of this strategy to determine in practice how much it really buys in terms of student achievement relative to other approaches. In other words, here is another promising direction which we have just begun to explore. It cannot be implemented in a book, and more needs to be done.

Most experimental psychologists reading the above discussion of drill and practice will find it difficult to suppress dark uncomplimentary mutterings about "1960's psychology". There are cycles in research, as in most things. In this dimension, we seem to oscillate between attacking small, tightly constrained, and fairly uninteresting problems over which we exercise a great deal of control, and attacking very large, sloppy, and interesting problems over which we can exert very little control. As may be evident from the above discussion and from reviews by Atkinson and Paulson (1972) and Fletcher (1975), drill and practice emphasizes the former. Nonetheless, it should also be evident that drill and practice is not just a matter of throwing items at students who are treated in some assembly line fashion. There are deep, educationally significant, and scientifically credible issues yet to be settled concerning drill and practice. Finally, it should be evident that despite the early strong results we have had from drill and practice, much more could be done to fully realize the promise of this approach.

As far as the oscillation between tightly controlled, less interesting problems and poorly controlled but much more interesting problems is concerned, it appears that current research in psychology, applied psychology, and instruction emphasizes the latter. This trend is especially apparent in current attempts to build tutorial dialogue systems. Nowhere is the attempt to automate single tutor/single student dialogue more evident. This is the line of development to turn to next.

Tutorial Dialogues

Before diving into the area of tutorial dialogues, a few comments on the automation of programmed textbooks may be in order. Most commentators on tutorial dialogue approaches include in this category the intrinsic programming techniques of Crowder (1959) that appear so frequently in commercially available computer instruction materials. Basically this approach uses the computer as a very large and sometimes very intricately programmed textbook. This is an approach that could be pursued in a book, although the book might have to be carried around in a wheelbarrow. Nonetheless, this approach appears to concern application of book and text technology rather than computer technology to instruction. It remains one of the most common, easily produced, and frequently implemented approaches, and it is best supported by authoring languages for computer instruction. The development of authoring languages such as PILOT, TUTOR, WISE, PLANIT, etc., all seem to have intrinsic programming in mind since this is the approach most easily taken when one uses these languages.

We tend not to publish our unsuccessful attempts at computer instruction, among other things, but there seems to be an underground consensus among those in the business that these intrinsic programming aproaches do not work very well. What appear to be intuitively obvious and correct procedures for assessing student knowledge, deciding when to branch, and providing remedial and accelerated material turn out to be relatively ineffectual in the light of student performance data. The determined reader is welcome to peruse Fletcher and Beard (1973) as an example of unpublished--and unsuccessful--work of this sort. In any case, this section does not concern the automation of programmed textbooks.

This section is concerned with the development of intelligent instructional systems as a new direction for computer instruction. This approach is a direct attempt to imbue computers with the qualities of expert human tutors. This line of development grew out of early concern with just how long it took, and how expensive it was, to generate items for computer presentation. Early estimates of the amount of time required to produce one hour of computer instruction ranged from 77 to 714 hours on PLATO, 200-400 hours on TICCIT, and around 475 hours for the IBM 1500 Instructional System (Orlansky & String, 1979). One solution to this problem was sought by those who noticed that the process of preparing items for computer presentation was boring, repetitious, and dull--in other words, a perfect job for computers. The resulting solution took the form of programs that would generate items for students (e.g. Koffman & Blount, 1974) and was called Generative Computer-Assisted Instruction, although what we now mean by generative computer instruction is a little more sophisticated. In any event, it occurred to early observers of the scene that since we were trying to use computers to mimic the item generation capabilities of expert human tutors, why not use computers to mimic all the capabilities of human tutors? Thus was born the notion of computerized tutorial dialogue.

The development of computerized tutorial dialogues involves the application of artificial intelligence techniques to computer instruction, resulting in the information structure oriented (ISO) approaches discussed and advocated by Carbonell (1970). Carbonell contrasted these approaches with ad hoc frame oriented (AFO) approaches based on techniques of programmed instruction. Carbonnell pointed out that, unlike AFO approaches, ISO approaches can be used to develop instructional systems that answer questions not specifically anticipated by the instruction designers, construct appropriate questions on given topics, and carry on a "mixed-initiative" dialogue in which either the student or the computer can introduce a response, topic, or idea in a free and comfortable subset of English. This may sound like programming a computer to be an expert tutor, and it is meant to.

This approach is in the mainstream of current developments in cognitive psychology which have taught us--or reminded us--that perception and learning are overwhelmingly constructive processes (cf. Resnick, 1983). In perception we do not collect bits of information from the "outside world" and paste them up on perceptual templates, and in instruction we are not writing information on blank slates in students' heads. Instead, we are dealing with active and very rich simulations of the world which students must create in order to perceive or learn. It is analysis by synthesis with a vengeance, and what gets transmitted in communication and instruction are not bits of information or knowledge but cues that may or may not be used to adjust the simulations being built up by students. The attempt in tutorial dialogue approaches is to deal directly with these simulations in ways that no drill and practice program--and no book--can.

Computers are both very good at this and very bad. Consider the following sentence:

The man the dog the girl owned bit died.

This is a difficult sentence for us to parse. We quickly become entangled in its syntactic nestings. Human chauvinism leads us to assume that since the sentence is difficult for us to parse, it is impossible for a machine. Yet a computer could quickly discern, after diving into its recursive routines for processing nested constructions, that there was a dog that was owned by a girl, that the dog bit a man, and that the man subsequently died.

Here is another example:

The man kicked the ball kicked the ball.

This is a perfectly grammatical sentence, as any self-respecting machine would discover after reversing an English transformational rule for deleting function words and determining that a man to whom a ball was kicked, kicked the ball back. In both these examples, a computer is less likely than we are to be confused or distracted, and ability to process these two examples illustrates real intellectual ability. "Artificial intelligence" is, after all, a poor name for the business of making computers intelligent. Intelligence, or intellectual ability, is really what the field is all about. That theories of intelligence are tested by algorithmization and putting them on computers is merely an issue of methodology, albeit a central one; there is nothing artificial about the capabilities targeted by this work.

Next we might consider the following example, taken from Donald Norman (1973):

What was Charles Dickens's telephone number?

A knowledgeable program would search the attributes it had associated with Charles Dickens looking for a telephone number, and/or it would search its list of telephone numbers looking for one associated with Charles Dickens. Presumably, no telephone number will be found for Charles Dickens, and this fact will simply be reported. As most human information processors must be aware, there is a more intelligent answer to this query that completely avoids searching lists of telephone numbers and/or attributes. To reply that telephones were not used until after Dickens's lifetime does require some knowledge, but this information could easily be stored by the computer. The real problem is to devise a computer program that knows when this information is relevant.

These examples also illustrate the distinction drawn by John Brown between "intelligent" computer instruction and what might be called "knowledgeable" computer instruction. In this light we might consider the following student/computer protocol taken from Brown's SOPHIE system (Brown, Burton, & Bell, 1974). In this example, the student is completing a series of tests on a computer simulated piece of electronic equipment (a regulated power supply) with a fault generated by the computer. Student reponses are underlined.

WHAT IS THE BSE EMITTER VOLTAGE OF THE VOLTAGE LIMITING TRANSITOR?

BASE

TRANSISTOR

THE BASE EMITTER VOLTAGE OF Q5 IS -.037 MILLIVOLTS.

The spelling corrector has detected and corrected two misspellings. The system must also know that Q5 is the voltage limiting transistor.

WHAT ABOUT THE CURRENT LIMITING TRANSISTOR

THE BASE EMITTER VOLTAGE OF Q6 IS .349 VOLTS.

The meaning of this question is completely dependent on the context established by the prior sentence. The system uses this context to determine that the student is interested not only in the voltage measurements but in the voltage across a particular junction.

WHAT SHOULD IT BE?

IN A WORKING CIRCUIT THE BASE EMITTER VOLTAGE OF Q6 IS .593 VOLTS.

Again the context determines the meaning of the question. Moreover, the system must infer that "should" implies what the measurement would be in a properly working device.

REPLACE Q6

I AM GOING TO ASK YOU SOME QUESTIONS ABOUT HOW Q6 IS FAULTED. ARE ANY JUNCTIONS SHORTED?

At this point SOPHIE knows that the student is barking up the wrong tree. Rather than simply replacing the transistor as requested, SOPHIE begins to query the student about what he or she thinks is wrong. If the student gets at least something right, the system says what was actually wrong and goes ahead and replaces the faulty part.

Clearly, SOPHIE has progressed beyond knowledgeable syntactic and semantic processing of student inquiries and is behaving in an "intelligent" manner. The system tries to create a reactive environment in which the student learns by actively trying out ideas in interaction with a simulation program. However, the program does more than simply simulate the information to be transmitted; it provides for tutorial feedback and, in effect, for a one-to-one relationship with an "articulate expert" problem solver who helps the student create, experiment with, and debug his or her own ideas.

Several reviews of this area have appeared, notable among which are discussions by Peele and Riseman (1975), Sleeman and Brown (1982), Barr and Feigenbaum (1982), and Fletcher (1984). Fletcher references about 16 of these tutorial dialogue systems that have been or are being developed. Carbonell's SCHOLAR (1970) and Brown's SOPHIE (Brown, Burton, & Bell, 1974) were seminal systems in the development of tutorial dialogues. The two premier systems currently seem to be GUIDON (Clancey, 1979) and Steamer (Williams, Holland, and Stevens, 1981).

GUIDON serves as a physician's consultant for the student, who plays the role of the physician, in diagnosing infectious diseases. GUIDON focuses directly on the problems a subject matter expert faces in making his or her expertise, understanding, and heuristics accessible to students. GUIDON takes account of students' knowledge and interests in choosing what to present, it incorporates a knowledge base that is augmented to better organize and explain the subject matter to the student, and its teaching expertise is represented explicitly and modularly so that it can be modified for different research designs. GUIDON both "knows" the subject matter and can explain to the student the paths it uses to reach a diagnosis just as an expert tutor does.

Steamer is a computer-based system being developed by the Navy to provide instruction in steam propulsion engineering. It links a very complicated and highly abstract, quantitative (mathematical) model of a ship's steam propulsion system to high quality visual (graphics) presentations of the underlying model. The student is thereby able to manipulate the underlying abstract model through the graphics interface and to see in computer graphics presentations how the effects of these manipulations would be propagated throughout the ship's steam propulsion system. Additionally, Steamer uses the student's manipulation to better model his or her understanding of steam and to extend, correct, and deepen that understanding.

At this point, we may all wonder if we are going to see tutorial dialogue systems of this sort in our classrooms in the near future. About a year ago one of the major figures in the tutorial dialogue world passed through Oregon State University leaving the following quote in his wake: "It's amazing what you can do when you only have two megabytes of memory."

To those of us used to working with 32K and 64K byte personal computers, the notion of 128K bytes seems like Nirvana. Two million bytes is beyond all imagining, and this is apparently the low end for someone working with tutorial dialogues. The point is that the computational requirements for tutorial dialogue systems are very large. A single user system sufficiently powerful for delivery but not development of tutorial dialgues might be purchased today for about $20,000. In ten years the picture will change completely, and for this reason the development of tutorial dialogue systems should now be pursued vigorously on large machines.

Somewhere among all the new directions for computer courseware a major breakthrough will occur. Tutorial dialogues appear to be a likely area for this breakthrough. This direction represents an approach that is both evolutionary and revolutionary. That is to say, we can expect it to help us accomplish what we want to do now and to alter in very fundamental ways our understanding of what instruction should be. In any event, tutorial dialogues could not be implemented without computers, and their development is limited by the current state of the art in both computer hardware and software. It is often said that hardware and software developments are far in advance of our capabilities to use them in instruction. In the case of tutorial dialogues, this is not true. We are simultaneously developing and capitalizing on the state of the art in computer hardware and software technology.

Much still needs to be done. We need to learn how to represent imperfectly understood and poorly described knowledge domains and to reduce the costs of creating knowledge domains. Better natural language processing must be developed, techniques for modeling learners must become far more sophisticated, and our understanding of what master tutors and teachers do must be greatly enhanced. We need to learn how to interface computer tutorial dialogues with the practice of classroom teachers. However, these issues only indicate that breakthroughs in this area will occur perhaps later rather than sooner. The promise of tutorial dialogues for improving instruction remains.

This promise is particularly evident when we review efforts to join tutorial dialogue techniques with simulation, the topic of the next section. In fact, we have already skirted these shoals very closely. After all, the student troubleshoots a simulated power supply in SOPHIE, diagnoses an ailing simulated patient in GUIDON, and operates a simulated steam propulsion system in Steamer. It may be past time to turn to the area of simulation in instruction.

Simulation

The currently strong and growing interest in simulation used for education is far overshadowed by the interest in and support for simulation used in training, specifically military and industrial training. Most readers will be familar with the long history and use of multi-million dollar aircraft simulators--some costing more than the aircraft they simulate--by the military and by aircraft manufacturers for pilot training. Twenty years ago, if one mentioned the use of simulators in instruction the reference would be to aircraft simulators and probably nothing else. The advent of computer technology has permanently altered this state of affairs.

Because current simulators are based on programmable computers, they need not be single purpose, representing only a single system such as the cockpit of an F-14 fighter aircraft. Instead, a wide range of related systems can be simulated for the purposes of training individuals who must learn to operate and maintain them. The Navy's Generalized Maintenance Trainer/Simulator (Rigney, Towne, King, & Moran, 1978) is a case in point. The GMTS can be used to simulate any device in which signal paths and their relationships to controls, indicators, and test points can be defined. So far the GMTS has demonstrated its versatility by being used to teach techniques to maintain both a radar repeater and a UHF communications systems.

Again because current simulators are based on programmable computers, they can be much smaller and less expensive than they were originally. Simulators too are benefitting from the micro-electronic revolution. The idea of "suitcase simulators" abounds in today's military. MITIPAC (Rigney & Towne, 1977), for instance, took the GMTS and shrunk it down via micro-electronics to fit into a suitcase-sized package which provides a true job site training capability. MITIPAC can now be transported to locations where military jobs are actually performed--in the field, on ships, on flight lines--and tailored to the specific jobs at hand. Many simulators have been built, tried, and evaluated in training, as Orlansky and String showed for training aircraft pilots (1977) and for training maintenance technicians (1981). In this sense, simulation is an established and proven technique for instruction. However, development of simulation for instruction is far from finished. The field is particularly fortunate in that promising and dramatic new "functionalities" now exist. Three of these new functionalities are interactive movies, surrogate travel, and spatial data management. All three of these use computer-controlled videodiscs.

Interactive Movies

Interactive movies attempt to translate movie viewing into an active, participatory process. In effect, the viewer becomes the director and controls many features of the movie. Feature controls available to the viewer are the following:

1. Perspective. The movie can be seen from different directions. In effect, the viewer can "walk around" ongoing action in the movie or view it from above or below.
2. Detail. The viewer can "zoom in" to see selected, detailed aspects of the ongoing action or can "back off" to gain more perspective on the action and simultaneous activity elsewhere.
3. Level of instruction. In some cases, the ongoing action may be too rich in detail or it may include too much irrelevant detail. The viewer can hear or see more or less about the ongoing process by so instructing an interactive movie system.
4. Level of abstraction. In some instances the viewer may wish to see the process being described in an entirely different form. For example, the viewer might choose to see an animated line drawing of an engine's operation to get a clearer understanding of what is going on. In some cases, elements shown in the line drawings may be invisible in the ongoing action, e.g., electrons or force fields.

5. Speed. Viewers can see the ongoing action at a wide range of speeds, including reverse action and still frame.

6. Plot. Viewers can change the plot to see the results of different decisions made at selected times during the movie.

Surrogate Travel

Surrogate travel forms a new approach to locale familiarization and low cost instruction. In surrogate travel, images organized into video segments showing discontinuous motion along a large number of paths in an area are stored on videodisc. Under microprocessor control, the student accesses different sections of the videodisc, simulating movement over the selected path.

The student sees with photographic realism the area of interest, for instance, a city street or a hallway in a building. The student can then choose both the path and the speed of advance through the area using simple controls, usually a joystick. To go forward the student pushes forward on the joystick; to make a left turn the student pushes the joystick to the left; to go faster the student pushes the joystick harder, and so on.

The videodisc frames the viewer sees originate as filmed views of what one would actually see in the area. To allow coverage of very large areas, the frames are taken at periodic intervals that may range from every foot inside a building, to every ten feet down a city street, to hundreds of feet in a large open area, e.g., a harbor. Coverage of very small areas is also of interest. In microtravel, which is a combination of surrogate travel and interactive movies, travel is possible where humans could could never go: inside watches while they are running, inside living organisms, etc.

The rate of frame playback, which is the number of times each video frame is displayed before the next frame is shown, determines the apparent speed of travel. Free choice in what routes may be taken is obtained by filming all possible paths in the area as well as all possible turns through all intersections. To some extent this is a time consuming and expensive technology, but it has become relatively efficient because of the design of special equipment and procedures for doing the filming.

Demonstrations of this technology have been developed for building interiors (National Gallery of Art), a small town (Aspen, Colorado), an industrial facility (nuclear power plant), and San Francisco Harbor. Plans are underway to produce a prototype video map library of broader scope for selected areas worldwide.

Spatial Data Management

Basically, spatial data storage and retrieval of information is the method of loci transformed to a video or computer graphics format. The information is stored and retrieved through its association with already familiar geographic terrain.

Suppose, for instance, a student wanted to study the musical environment in which Ralph Vaughan Williams wrote his "Concerto for Tuba and Orchestra". In an ordinary data retrieval system the student will type in a complicated set of Boolean expressions--or English phrases standing for Boolean expressions--and will receive in return only textual information about the topic. Relevant information closely related to the information successfully retrieved will not appear unless the student starts from the top again with a new set of Boolean expressions. In a spatially organized data system, the underlying geography will be familiar to the student, for instance the school campus. The student may then "fly" to the music department (or library, concert hall, professor's office, etc.) and look for a tuba (or an orchestra, music library, portrait of the composer, etc.). Upon finding a tuba or other relevant cue, the student can "zoom" into it, still using his single joystick contol, select the concerto by name (or by hearing it, seeing the score, seeing the composer, etc.) and then hear, see, and read more information about it all retrieved through visually oriented associations.

In this way, spatial data management acts as an electronic library that gives students and instructors access to a wide assortment of multi-source and multi-media information whose components are associated in a natural and easily accessible manner. Instructors can access the system to create and/or assemble their own information spaces to be explored later by their students or subsequently present these materials to large audiences in single locations using large screen television projection or to multiple locations though cable distribution systems. Students can independently use the system for individualized instruction by working though previously designed information spaces, by browsing on their own, or by creating their own data spaces. When students and instructors are in remote locations, offsite instruction can be facilitated by linking two or more systems together using regular telephone lines. In this manner, a student or instructor can "fly" the other to a topic of interest, sharing at geographically remote sites a large, visually oriented library of information.

Two points are worth noting about these new directions for simulation applied to instruction. First, they cannot be implemented in a book. Second, the application of these new directions for simulation-based computer instruction in education is just beginning. One can easily imagine application of this technology to science education. Perhaps a few words on this subject are in order.

The best way to learn science is by doing it. The excitement, mystery, frustrations, and triumphs of science are only dimly revealed by the usual fare of introductory science course. It would be far better for students, especially introductory students, to approach science with freedom to indulge their curiosity, form and re-form their own hypotheses, design and perform their own experiments, and build their own models and theories to explain natural phenomena. Unless there are drastic shifts in national funding policies for science education, this essential scientific experience will be prohibitively expensive to provide. The result is that students--especially elementary and junior high school students--are "turned off" by science at a time when our industrial and academic need for scientists, engineers, and technologists is acute and increasing.

What is needed in science education is something that has the impact of video gaming, but at the same time possesses substantial pedagogical power. One way to accomplish this is to provide simulated scientific experiences to students. Good simulations are exciting, compelling, and teach effectively by providing an environment in which learners must live with their decisions. Simulated experiences need not replace existing laboratory and field exercises, but they may expand and supplement them. Moreover, simulated experiences may be superior to real experiences in at least four ways. First, and primarily, simulation can be economical. Use of simulation should reduce the need for laboratory equipment and its maintenance, laboratory supplies, and travel costs for field experience. Second, simulation can make relevant phenomena more readily visible in two ways. In one way it can make the invisible visible. For instance, the flow of ions can be seen more clearly and simply under simulated conditions than under real conditions. In another way, simulation may increase the visibility of a phenomenon by separating it from a confusing or chaotic background. One can see the conceptual forest without getting lost in the procedural trees. Third, simulation allows reproducibility. Students can replay over and over chains of events that they could not otherwise observe repeatedly. Fourth, simulated experience is often safer than the real thing. Airplanes can be crashed, poisons can be ingested, and laboratories can be exploded with impunity in simulated environments.

Two sorts of relevant scientific experience that lend themselves readily to simulation are field study and laboratory experimentation. These two kinds of experience could be provided using the new functionalities described above. These functionalities could be used to build video field trips and simulated laboratories.

In the field, the student sees the total ecological view. He/she sees the overall landscape, the terrain, the populations of organisms, and individual samples of interest in their special areas. In sciences such as biology, geology, paleontology, archeology, and even astronomy, substantial learning and appreciation can be achieved by travel to locations that are difficult to access under the best of conditions. However, field trips are treated as an instructional frill. After all, the trips are made rarely and locally (they depend for success on what is serendipitously nearby); they emphasize only the group (individuals do not have an opportunity to do the science on their own); and most of the administrative effort centers on getting to the field and getting back, not on the field experience itself. As a result, even short, local field trips are being cancelled by schools because their cost in time and fuel is not balanced by their educational return. Surrogate travel removes the major objections to field experience and offers to each student a broadened opportunity to experience scientific phenomena in their natural, ecological context.

Students interested in, say, the biology of deserts could visit the Gobi in the morning, the Sahara around noon, and the Sonoran in the afternoon. They could travel around in each habitat locating, identifying, and "gathering" samples in roughly the same way, and for the same purposes, as a trained scientist. Panning and zooming through the full range of habitats could develop in students many of the same intuitions and understandings of environmental, geographic, and climatic contexts that an experienced scientist gains from actual travel.

Back in school, laboratories provide a problem solving environment where students interact, observe processes, and are stimulated to synthesize concepts as part of their learning. However, many schools are eliminating laboratories from their science courses, not because they are not useful learning experiences, but because of the cost of obtaining, maintaining, and supporting specimens, samples, and laboratory equipment. Interactive movies and spatial data management allow us to simulate laboratory experiences without the high cost and effort that is normally involved under the present pattern.

Students can create, store, and retrieve information from mammoth data banks using spatial data management. One can imagine high school students organizing an entire archaeological excavation or geological survey using spatial data techniques. One can also imagine elementary school students setting up and running high-energy particle physics experiments through interactive movies with plot control. Students would also have full use of the latest in telescopes, microscopes, and even endoscopes through computer-based simulation.

Finally, laboratory and field experiences can be linked so that hypotheses developed in the laboratory would be tested by return "travel" to the correct habitat, "collection" of data or specimens, and return to the laboratory for testing and verification. In this way, the excitement, frustrations, and triumphs of scientific experiences would become accessible to students.

In the above, simulation was presented as a new direction that is finding its way into computer instruction, but it is interesting to note that the history of computer instruction is exactly the reverse.

The first use of computers to teach grew out of a computer based system that was primarily intended for simulation of real world experiences. This was the Air Force's SAGE (Semi-Automatic Ground Environment) system which was built in the late 1950's to train Air Force personnel in the techniques and tactics of air defense (Rowell & Streich, 1964; Parsons, 1972). Computers in SAGE were inititally used to simulate equipment, mostly radar, to which ground based air defense personnel were to make appropriate reactions. However, as time progressed, the SAGE computers began to be used to present training in a more general-purpose fashion.

The University of Illinois's PLATO (Programmed Logic for Automatic Teaching Operations) was probably the first computer system built specifically for computer instruction. Interestingly, it too was first supported solely by the military--in this case by the Army Signal Corps, the Office of Naval Research, and the Air Force Office of Scientific Research (Bitzer, Braunfeld, & Lichtenberger, 1962). Initially PLATO was used as a sort of "book with feedback" following the suggestion of Chalmers Sherwin, and few who saw early demonstrations of PLATO in the late 1960's were able to escape its "fruit fly" demonstration. This was a simulated biology laboratory showing in high quality graphics successive generations of fruit flies as they illustrated a model of genetics. This type of simulation in computer instruction is still in use.

The focus in this section is on new techniques for simulation, three of which are listed above. These three have been discussed in a little more detail by Bolt (1979) and by Levin and Fletcher (1981).

Other techniques may well be on the way. We have barely begun to explore the instructional possibilities of natural language processing (as opposed to computer language processing), voice output, voice input, computer-generated imagery (which may obviate some of the need for videodisc storage), and psychoneurological monitoring. New functionalities for these capabilities will doubtless be developed. However, it should be emphasized that this process of discovery is at least as demanding of time, resources, and ingenuity as the development of the computational capabilities themselves. Swamping schools with hardware and computer capabilities and then expecting instructional functionalities to flow spontaneously in their wake is simply wrong. The process will continue to require support, encouragement, resources, and time.

Final Word

It is wrong to inundate our educational institutions with new technologies without insisting that they do at least something to help us through the day. It is also wrong to hold off all investment in new technologies because they may affect what it is we want to do. The correct approach seems to be somewhere in the middle. No one envisioned teleconferencing when the telephone was invented, no one imagined our current interstate highway transportation system when the horseless carriage came along, and steam engines languished for 30 years pumping water out of coal mines before someone began to think seriously of their possibilities for self-locomotion. We have benefitted from the introduction of these devices into our lives just as we have suffered from them. We must give the new technologies their place if we are to improve our instructional practice as the Gardner Commission said we must. At least in the case of computers, we are in a position to insist that they be of some immediate practical value along the way. This is a fortunate position to be in, and we should capitalize on it. Computers can help meet goals and solve current problems of schools and school districts at the same time they are helping to advance the craft of instruction. We can and should expect them to do both.

In short, computers will help us better perform the business of instruction as we envision it today. They will also broaden our horizons. They will change and expand our ideas about what instruction is and what it must do. Their challenge to us as educators is as serious as their promise. We should rise to the occasion.

References

Atkinson, R.C., & Paulson, J.A. (1972). An approach to the psychology of instruction. Psychological Bulletin, 78, 49-61.

Barr, A., & Feigenbaum, E.A. (1982). Handbook of artificial intelligence: Vol. 2. Los Altos, CA: William Kaufman.

Bitzer, D.L., Braunfeld, P.G., & Lichtenberger, W.W. (1962). PLATO II: A multiple-student, computer-controlled, automatic teaching device. In J.E. Coulson (Ed.), Programmed learning and computer-based instruction. New York: John Wiley and Sons.

Bolt, R.A. (1979). Spatial data management. Cambridge, MA: Massachusetts Institute of Technology.

Brown, J.S., Burton, R.R., & Bell, A.B. (1974). SOPHIE: A sophisticated instructional environment for teaching electronic troubleshooting (An example of AI in CAI) (BBN Report 2790). Cambridge, MA: Bolt Beranek and Newman, Inc.

Bunderson, C.V. (1981). Courseware. In H.F. O'Neil, Jr. (Ed.), Computer-based instruction: A state-of-the-art assessment. New York: Academic Press.

Carbonell, J.R. (1970). AI in CAI: An artificial intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems, 11, 190-202.

Chant, V.G., & Atkinson, R.C. (1973). Optimal allocation of instructional effort to interrelated learning strands. Journal of Mathematical Psychology, 10, 1-25.

Clancey, W.J. (1979). Tutoring rules for guiding a case method dialogue. International Journal of Man Machine Studies, 11, 25-49.

Crowder, N.A. (1959). Automated tutoring by means of intrinsic programming. In E.H. Galanter (Ed.), Automatic teaching: The state of the art. New York: John Wiley and Sons.

Fletcher, J.D. (1975). Models of the learner in computer-assisted instruction. Journal of Computer-Based Instruction, 1, 118-126.

Fletcher, J.D. (1979). Computer-assisted instruction in beginning reading: The Stanford projects. In L.B. Resnick & P.A. Weaver (Eds.), Theory and Practice of Early Reading: Vol. 2. Hillsdale, NJ: Lawrence Erlbaum Associates.

Fletcher, J.D. (in press). Intelligent instructional systems in training. In S.A. Andriole (Ed.), Applications in artificial intelligence. New York: Petrocelli Books.

Fletcher, J.D., & Beard, M.H. (1973). Computer-assisted instruction in language arts for hearing-impaired students (Technical Report No. 215). Stanford, CA: Institute for Mathematical Studies in the Social Sciences, Stanford University.

Gilbert, T.F. (1960). On the relevance of laboratory investigation of learning to self-instructional programming. In A.A. Lumsdaine & R. Glaser (Eds.), Teaching machines and programmed learning: A source book. Washington, DC: National Educational Association.

Koffman, E.B., & Blount, S.E. (1974). A modular system of generative CAI in machine-language programming. IEEE Transactions on Systems. Man, and Cybernetics, 4, 335-343.

Laubsch, J.H. (1970). Optimal item allocation in computer-assisted instruction. IAG Journal, 3, 295-311.

Levin, S., & Fletcher, J.D. (1981). Videodisc technology: An approach to the design of training devices. In J. Moraal & K.F. Kraiss (Eds.), Manned systems design: Methods, equipment, and applications. New York: Plenum Press.

Lorton, P.V. (1973). Computer-based instruction in spelling: An investigation of optimal strategies for presenting instructional material. Doctoral dissertation, Stanford University, Stanford, California.

McLuhan, H.M., & Fiore, Q. (1967). The medium is the message. New York: Random House.

Orlansky, J. & String, J. (1981). Cost-effectiveness of maintenance simulators for military training (IDA Paper P-1568). Arlington, VA: Institute for Defense Analyses.

Papert, S. (1980). Mindstorms. New York: Basic Books.

Parsons, H.M. (1972). Man-machine system experiments. Baltimore, MD: Johns Hopkins Press.

Paulson, J.A. (1973). An evaluation of instructional strategies in a simple learning situation. Doctoral dissertation and Technical Report No. 209. Institute for Mathematical Studies in the Social Sciences, Stanford University, Stanford, California.

Peelle, H.A., & Riseman, E.M. (1975). Four faces of HAL: A framework for using artificial intelligence techniques in computer-assisted instruction. IEEE Transactions on Systems, Man, and Cybernetics, 5, 375-380.

Resnick, L.B. (1983). Mathematics and science learning: A new conception. Science, 220, 477-478.

Rigney, J.W., & Towne, D.M. (1977). Some concepts, devices, and procedures for improving technical training and on-the-job performance of maintenance (Technical Report 76-C-0023-1). Orlando, FL: Naval Training Equipment Center.

Rigney, J.W., Towne, D.M., King, C.A., & Moran, P.J. (1978). Field evaluation of the generalized maintenance trainer-simulator: I. Fleet communications system (Technical Report No. 89). Los Angeles, CA: Behavioral Technology Laboratories, University of Southern California.

Rowell, J.T., & Streich, E.R. (1964). The SAGE system training program for the Air Defense Command. Human Factors, 6, 537-548.

Rumelhart, D.E. (1967). The effects of interpresentation intervals on performance in a continuous paired-associate task (Technical Report No. 116). Stanford, CA: Institute for Mathematical Studies in the Social Sciences, Stanford University.

Sleeman, D., & Brown, J.S. (Eds.). (1982). Intelligent tutoring systems. New York: Academic Press.

Suppes, P. (1964). Problems of optimization in learning a list of simple items. In M.W. Shelley & G.L. Bryan (Eds.), Human judgments and optimality. New York: John Wiley and Sons.

Suppes, P. (1967). Some theoretical models for mathematics learning. Journal of Research and Development in Education, 1, 5-22.

Suppes, P., Fletcher, J.D., & Zanotti, N. (1975). Performance models of American Indian students on computer-assisted instruction in elementary mathematics. Instructional Science, 4, 303-313.

Suppes, P., Fletcher, J.D., & Zanotti, N. (1976). Models of individual trajectories in computer-assisted instruction for deaf students. Journal of Educational Psychology, 68, 117-127.

Williams, M., Hollan, J., & Stevens, A. (1981). An overview of STEAMER: An advanced computer-assisted instruction system for propulsion engineering. Behavior Research Methods and Instrumentation, 13, 85-90.