DOCUMENT RESUME

ED 237 064 IR 010 888

AUTHOR	Jones, Nancy Baker, Ed.; Vaughan, Larry, Ed.
TITLE	Evaluation of Educational Software: A Guide to Guides.
INSTITUTION	Northeast Regional Exchange, Inc., Chelmsford, MA.; Southwest Educational Development Lab., Austin, Tex.
SPONS AGENCY	National Inst. of Education (ED), Washington, DC.
PUB DATE	Jan 83
CONTRACT	400-80-0107
GRANT	NIE-G-82-0017
NOTE	102p.
PUB TYPE	Guides - Non-Classroom Use (055) -- Tests/Evaluation Instruments (160)
EDRS PRICE	MF01/PC05 Plus Postage.
DESCRIPTORS	Check Lists; *Computer Programs; Evaluation Criteria; *Guidelines; *Information Sources; *Instructional Materials; Media Selection; *Models; Publications; Rating Scales
IDENTIFIERS	Software Evaluation; *Software Reviews

ABSTRACT

This guide suggests processes for making informed software selection, describes several prominent software evaluation models, raises key issues in the software selection process, and provides information on a variety of print and organizational resources. The first section, featuring an article by Henry F. Olds, Jr., "Evaluating the Evaluation Schemes," presents several key principles for educators to consider in software evaluation. The second section contains brief summaries of ten software evaluation models from organizations representing both non-profit, cooperative systems, and private companies: (1) MicroSIFT (Microcomputer Software and Information for Teachers); (2) EPIE (Educational Products Information Exchange) and Consumers Union; (3) "School Microware Reviews"; (4) "Courseware Report Card"; (5) Minnesota Educational Computing Consortium (MECC); (6) SOFTSWAP (San Mateo Educational Resources Center); (7) CONDUIT; (8) National Council of Teachers of Mathematics; (9) Scholastic Book Services; and (10) "Electronic Learning." Several of the summaries describe overall strategies for software evaluation, procedures, criteria, and rating systems. The third section includes sample software reviews that show the results of applying criteria and ratings on various dimensions of educational software. The final section provides several resource lists focusing on various aspects of software availability and evaluation. The listings include books, journal articles, periodicals, directories, and other resources. (LMM)

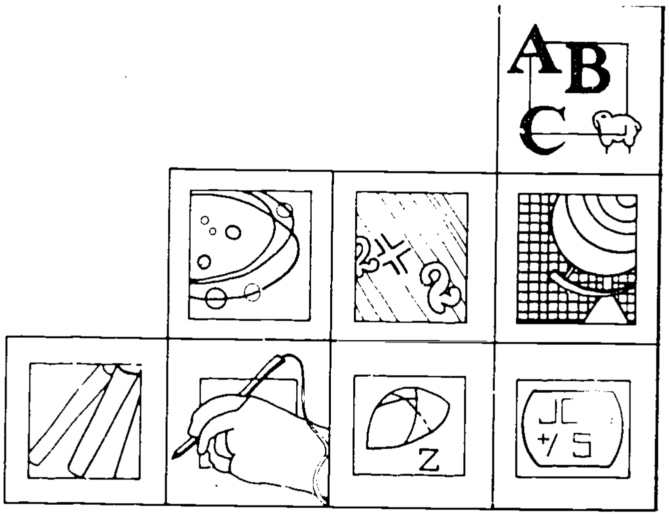
Evaluation of Educational Software

a guide to guides

Edited by

Nancy Baker Jones
Regional Exchange
Southwest Educational
Development Laboratory
Austin, Texas

Larry Vaughan
Northeast Regional Exchange, Inc.
Chelmsford, Massachuset**



The publication was developed by the Northeast Regional Exchange, Inc. and the Southwest Educational Development Laboratory. The development was supported by two separate awards from the National Institute of Education, U.S. Department of Education (NIE G-82-0017 and NIE 400-80-0107). The opinions expressed in this publication do not necessarily reflect the position or policies of the U.S. Department of Education, the Northeast Regional Exchange, and the Southwest Educational Development Laboratory. Mention of trade names, commercial products, or organizations does not imply endorsement by the United States Government or the publishers of this document.

THE NORTHEAST REGIONAL EXCHANGE

The Northeast Regional Exchange, Inc. (NEREX), a private, not-for-profit corporation, is a service agency that promotes educational equity and improvement. NEREX coordinates resources and sharing of information among the seven State Departments of Education in the Northeast based on an established set of state and regional priorities. Through NEREX, states are able to expand their available resource base and work through regional sharing efforts toward program improvement in local school districts and other educational institutions. The Northeast Regional Exchange, Inc. is governed by a Board of Directors that includes the seven Chief State School Officers from the Northeast and eight representatives from a wide variety of education constituency groups in the region.

BOARD OF DIRECTORS

Connecticut:	David G. Carter Mark R. Shedd, Chair
Maine:	Robert Goettel Harold Raynolds, Jr.
Massachusetts:	Kevin T. Andrews John R. Lawson
New Hampshire:	Robert L. Brunelle Jeannette Vezeau, C.S.C.
New York:	Gordon M. Ambach Anne L. Bondy, Vice-Chair Sandra Feldman
Rhode Island:	Arthur R. Pontarelli Ann Prosser
Vermont:	Stephen S. Kaagan Patricia M. Zyber
Treasurer:	Richard J. Lavin

Executive Director:
J. Lynn Griesemer

THE SOUTHWEST EDUCATIONAL DEVELOPMENT LABORATORY

Regional Exchange Project

The mission of the Southwest Educational Development Laboratory (SEDL), as defined by its Board of Directors, is to promote quality education by conducting behavioral and social research, by engaging in educational development, and by providing technical assistance to education-related institutions. SEDL is a non-profit corporation chartered in 1966 in the state of Texas. Activities are concentrated in, but not limited to, six states: Arkansas, Louisiana, Mississippi, New Mexico, Oklahoma, and Texas. The Regional Exchange at SEDL (SEDL/RX) promotes educational improvement efforts through the dissemination and use of educational research and development. With the guidance of an Advisory Board whose members represent the six state education agencies and the Office of the Secretary's Regional Representative in the SEDL region, the SEDL/RX supports states in meeting their own priorities by providing syntheses of research, seminars and symposia, on-site assistance, and access to a variety of research-related resources throughout the United States.

BOARD OF DIRECTORS

Arkansas:	Roland Carpenter Joseph K. Mahony, II Jane Mendel
Louisiana:	Cleo A. Anders Sylvia Blanchard Mark C. Surprenant
Mississippi:	Yvonne Brooks Tom Cheney Leslie D. King
New Mexico:	Orlando Giron Julia Herrera Maurice Hobson
Oklahoma:	James L. Casey Anita Chisholm Earl Nichols
Texas:	Eli Douglas George Kozmetsky Valleau Wilkie, Jr.

Executive Director:
Preston C. Kronkosky

Foreword

Computer instruction has become a regular feature in classrooms across the country. The successful use of computers to augment classroom instruction requires that teachers learn several new skills. In particular, educators are seeking help in the identification of software that can best serve their needs. For this reason, the Regional Exchange at the Southwest Educational Development Laboratory, (SEDL/RX) and the Northeast Regional Exchange, Inc. (NEREX) have joined forces to produce a resource book about software evaluation. By collaborating in the development of this book, not only have both organizations benefited, but combining the resources of two regions has greatly expanded the scope of available information about software evaluation. The editors, Nancy Baker Jones of the SEDL Regional Exchange and Larry Vaughan of the NEREX staff, deserve praise for the result of their collaboration. In their work, the editors successfully engaged the talents and perspectives of several key education professionals in both regions, as well as from the national Research and Development Exchange network (RDx).

The Northeast Regional Exchange, and the Regional Exchange of the Southwest Educational Development Laboratory are indebted to many people for their assistance in the development of this book. We are particularly grateful for the assistance of the members of the NEREX Instructional Technology Task Force and the SEDL/RX Advisory Board. The Task Force helped to mold the overall design for this book and contributed much of the resource information in the appendices, while the SEDL/RX Advisory Board provided the impetus for SEDL's involvement in producing this book and provided support throughout the process. Members of the NEREX Task Force are:

> Elizabeth Glass, Connecticut; Richard Riley, Maine; Roselyn Frank, Paul Maloney, Clark Adams, and Susan Foote, Massachusetts; Fernand Prevost, New Hampshire; Norman Kurland, Robert Trombly, and Greg Benson, New York; Harry Darling, and Robert Rude, Rhode Island; and Robert Kenney, Vermont.

Members of the SEDL/RX Advisory Board are:

> Sara Murphy, Arkansas; Sue Wilson, Louisiana; Clyde Hatten, Mississippi; Susan Brown, New Mexico; Jack Craddock, Oklahoma; Marj Wrightman, and J.S. Miguel, Texas.

Henry F. Olds, Jr., author of "Evaluating the Evaluation Schemes," contributed to the development of this book by providing information on resource articles, linkage with key people, and insights about the key issues involved in software evaluation.

Many educators in our respective regions have generously given their time to share their concerns for software evaluation. We are grateful to them and to all who so generously gave their time and materials for inclusion in this publication.

The editorial board for *Evaluation of Educational Software: A Guide to Guides* is also due applause. Its members — Vicki Blum Cohen, Director of Instructional Design for ISO Communications, Inc., and formerly with EPIE and Columbia University's Microcomputer Resource Center; Donald Holznagel, Coordinator of MicroSIFT, Northwest Regional Educational Laboratory; and James Poirot, Chair of the Department of Computer Sciences, North Texas State University — assisted the editors by recommending the best resources from the rather large world of information on software evaluation. They also reviewed and commented on the final draft of the book.

For production of this book, we owe thanks to Gloria Bowles of the NEREX staff, who was involved in development and refinement of this book from the beginning. Dennis Collins of LEA Associates in Concord, New Hampshire, was instrumental in preparing the document for printing, and Desireé Burke, an Austin artist, designed the cover.

Preston C. Kronkosky
Executive Director
SEDL

J. Lynn Griesemer
Executive Director
NEREX

Contents

Foreword V

Introduction 1

Evaluating the Evaluation Schemes"
by Henry F. Olds, Jr. 3

Approaches to Software Evaluation 11

MicroSIFT 12

EPIE & Consumers Union 16

School Microware Reviews 29

Courseware Report Card 32

Minnesota Educational Computing Consortium 34

SOFTSWAP 41

CONDUIT 46

National Council of Teachers of Mathematics 54

Scholastic 62

Electronic Learning 65

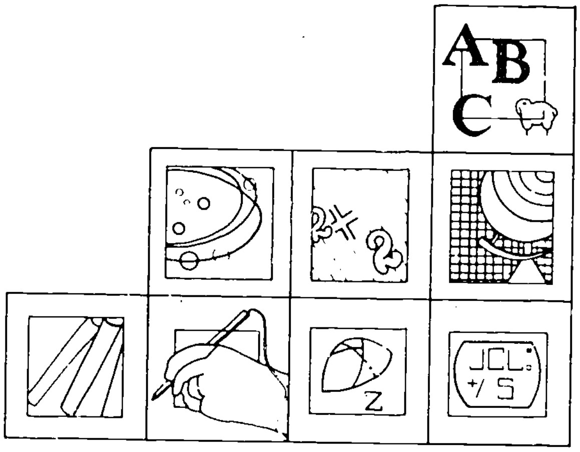
Software Reviews 69

MicroSIFT 70

School Microware Reviews 71

Courseware Report Card 73

Resource Information 77



Introduction

Current surveys from virtually every section of the country reveal a variety of efforts to augment classroom instruction through the use of computers. Although a few schools have been using computer instruction for more than a decade, many schools have just begun, now that the microcomputer has made computer instruction more feasible. The number of microcomputers in schools is growing at a rate approaching geometric progression. Educators face a variety of problems as they attempt to integrate computer instruction in schools. In particular, there are important decisions on the allocation of limited resources for computer instruction.

The selection of software to support computer instruction often represents a considerable investment, and these investment decisions continue to be complicated by the dynamics of a rapidly developing field. Good information is needed by educators to help guide the software selection process. More and more information about software is becoming available from a variety of different sources, but it is difficult to locate and keep track of reliable sources of information about the selection of software.

Our overall goal in developing *Evaluation of Educational Software: A Guide to Guides* is to provide awareness, understanding, and reference information about the evaluation of software. Specifically, this document is intended to:

- present suggested processes for making informed software selections
- provide awareness of several prominent models for software evaluation
- raise key issues in the software selection process, and
- provide information on a variety of print and organizational resources that may be useful to those making software decisions.

This book is designed to serve several educational audiences. Individual administrators and teachers who have responsibility for software selection, members of school-based courseware selection committees, pre-service and inservice students who are forming opinions about software, and educators who are developing their own software can use this publication in a variety of ways. Information in the following pages may help guide the development or refinement of a local philosophy and policies regarding software evaluation. Descriptions of software evaluation models and systems may be used to help determine what sources of information should be regularly accessed to inform local decision making. The procedures, criteria, and forms from these evaluation models can be adapted for local applications. Also, the reference materials and organizations cited can guide educators as they seek specific types of information about software availability and evaluation. We hope that commercial software developers will make use of the information contained in this book as they develop new instructional software.

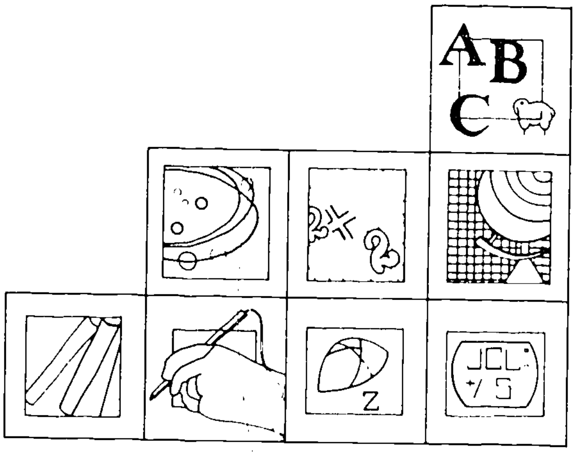
This book has four main sections. The first section, featuring an article by Henry F. Olds, Jr., "Evaluating the Evaluation Schemes," presents several key principles for educators to consider in the evaluation of software. The final responsibility for selecting computer software rests with local decision makers. These educators have the responsibility for knowing what constitutes appropriate software for various instructional computing needs. The major organizations engaged in software evaluation described in this book provide useful services. These organizations have identified a variety of issue categories pertinent to software evaluation, developed evaluation criteria, developed review and evaluation forms that can be adapted for local uses, and provided descriptive and evaluative comments about specific software packages. We recommend that educators not rely too heavily on software evaluation conducted by persons outside of the local school system. No evaluation service or form, by itself, can serve the real purpose of evaluating software: to find pedagogically sound curricula materials appropriate for the learner and the lesson. Published reviews and evaluations help to identify software that can be more carefully examined at the local level in light of local criteria.

The second section of this publication contains brief summaries of ten major models for software evaluation. The ten organizations included represent both non-profit, cooperative systems, and private companies. Several of the summaries describe overall strategies for software evaluation, procedures, criteria, and rating systems. We recommend that interested educators find out more about these organizations and adapt those procedures, criteria, and strategies that appear most in line with their own software evaluation priorities.

The next section includes sample software reviews. These show the results of applying criteria and ratings on various dimensions of educational software. The final section of this book contains several resource lists focusing on various aspects of software availability and evaluation. New, pertinent print and organizational resources are becoming available as the field of computer instruction expands and develops. The books, journal articles, periodicals, directories, and other resources listed here were valuable to the editors in developing this publication. We recommend that educators investigate these and other resources as they develop the local capacity to review, evaluate, and select educational software.

Evaluating the Evaluation Schemes

by Henry F. Olds, Jr.



Toward a New Perspective on Software Evaluation

The value of any computer lies in the value of the software available for it. While the microprocessor has made computers affordable for use in homes and classrooms, purchasing a microcomputer or, more often, several microcomputers, is still a substantial investment. That investment is relatively worthless without the software that makes the microcomputer useful. Indeed, the software investment required to use the computer may exceed the original investment in the computer itself.

Now that there is increasingly more sophistication about how this technology can be appropriately applied to education, we know that a very large percentage (some say as high as 90%) of educational software currently on the market is not worth buying. Much of it consists of misguided efforts to translate textbook materials into software. In other cases, the software is poorly conceived, poorly designed, and difficult to use. Software that combines the best efforts of experienced educators and expert programmers is only just beginning to reach the market in any quantity.

There are a few general features that characterize quality software:

1) It should be the outgrowth of a fully conceived and carefully articulated "intellectual model of the content domain" (e.g., good science software should represent the best current scientific understanding).

2) It should reflect an understanding of the cognitive developmental needs and capacities of the learner (e.g., younger children need more concrete representational experiences).

3) Since the computer provides an environment for interaction, its best pedagogical use should support the inherently interactive nature of knowledge construction for the learner (e.g., the computer is a poor medium for giving a lecture).

4) It should make use of the special qualities of computer technology in truly functional ways (e.g., use graphics to make an abstract concept concrete). It should not attempt to carry out instructional tasks far better suited to other media (e.g., it should not be a textbook).

Each of these features can be much further refined. While the first two also apply to non-computer instructional materials, they must be reconsidered in the light of this new technology. The last two are the areas where most current attention is now focused.

The size and importance of the investment in computers and the lack of quality software make choosing good software a critical issue for educators today. This issue is complicated by the lack of a shared language to communicate about a new technology that has the capacity to provide a highly multi-sensory experience in a thoroughly interactive fashion. Few educators yet are able to talk knowledgeably about this new mode of instruction with which they have had almost no direct or extensive experience. Unfortunately, the experiences of most have been indirect (hearing or reading about computing, not doing it) and limited to only a few moments with only one kind of software.

Nevertheless, the question we must concern ourselves with is: how can people be helped to make good choices in a new domain where there is little shared experience and no shared language? One response to this question is to create software evaluation forms that can be filled out by experienced computer-using educators. The evaluations can then be synthesized and reported to others seeking useful information in a way that contributes to making good software choices.

On the surface, this approach sounds reasonable and worthwhile. Very serious, well-intentioned efforts have been made to develop evaluation formats that are both informative and useful, and these efforts merit our attention. A number of them are presented in this book. They are informative both because they lay out much of the domain that must be considered and because they illustrate many of the dilemmas inherent in this form of evaluation.

Any form of evaluation might be expected to have the following features:

1) Since the primary function of a good evaluation is to inform, software should be described in detail before it is evaluated.

2) Evaluation should be qualitative, not quantitative. Good and bad features should be discussed, and rating scales should be kept to a minimum. Numbers should not be relied upon to convey important substantive matters.

3) Evaluations should be timely; they should treat recently published products shortly after initial publication.

4) The evaluation should compare the program under consideration with others of a similar type.

5) Evaluations should distinguish between those aspects of design that may be easily remedied and those which cannot be changed without overall reconceptualization.

Description Before Evaluation

In *Mindstorms*, Seymour Papert says, "The computer is the Proteus of machines. Its essence is its universality, its power to simulate. Because it can take on a thousand forms and serve a thousand functions, it can appeal to a thousand tastes.

The thousands of forms that a computer can become are the products of the software created for it. To understand the computer, therefore, is to be able to describe its software. The most important task that must be undertaken, before attempting the evaluation of any particular program, is to create a general descriptive framework for software that can be used to make as clear as possible what a software program is and what it is not.

Just as it does not make any sense to evaluate a meal of beef bourguignonne at a fine French restaurant in the same terms as a hamburger at a fast food outlet, even though they can both be called food, it does not make sense to evaluate a drill and practice program on math facts in the same terms as a database program, even though they are both software. It is terribly important that people begin to understand that software is probably no less diverse than food. Furthermore, there is "junk software" — software of dubious merit that should, perhaps, have a warning attached to it about possible "health hazards," and there is "gourmet software" — perhaps beyond the average person's everyday tastes or needs but which, nonetheless, sets high standards of quality for all other software.

A commonly accepted way of describing software (a descriptive taxonomy) is badly needed. It must not be a rigid framework because software development is still in its infancy, and any framework must be adaptable to new (and undoubtedly surprising) creations that don't quite fit. Several years ago, the Cognitive Research Group at Education Development Center (Newton, MA) developed a beginning framework as part of an intensive study of computer software supported by Control Data Corporation (Olds, Schwartz and Willie, "People and Computers: Who Teaches Whom?"). As a regular software reviewer for two publications (*Classroom Computer News* and *WINDOW*), I have found this framework extremely helpful in understanding how to think about software and in describing it to my readers. As a trainer of teachers, I have found this framework of tremendous value for developing a shared language to talk about software.

Over a couple of years, I have modified the original framework slightly and changed the language a bit here and there. The framework places software somewhere on a continuum: from plain and unsophisticated applications (fast food) to fancy and highly sophisticated applications (gourmet). It must be emphasized that this is not an evaluative framework — just as there can be very good and healthy fast food, which can be just the right food on certain occasions, so there can be very good and effective software that is very plain and unsophisticated.

A Descriptive Framework

It is helpful to distinguish between three broad categories of software, though in reality the distinctions sometimes get a bit fuzzy at the boundaries. First, there is software that uses the computer as a medium to transmit information or to instruct (often referred to as CAI — computer assisted instruction). Then there is software that uses the computer as a modeling device for creating an environment with which the user interacts. And finally, there is software that uses the computer as a tc~l with which the user performs some task. Each of these major categories has some important sub-categories, which we shall now describe briefly.

I. Computer as Instructional Medium (CAI)

A. *Drill and Practice* — the use of the computer to evoke continued and improved performance in some well-specified skill or knowledge domain. The computer can provide potentially limitless practice exercises and immediate feedback to the user. Effective applications of drill and practice software have been made in math number facts, spelling, foreign language vocabulary, typing, etc.

B. *Tutorials* — the use of the computer to teach some subject matter directly. At their best, some tutorials use the computer to respond to the user by adjusting both the feedback and the continuing instruction to the user's growing understanding (a process known as branching). A fast learner moves quickly through the material, while a slower learner is given several alternative opportunities to learn. As with drill and practice, tutorials tend to be most effective when the content being taught can be very clearly specified. They also work best when a high level of motivation can be assumed on the part of the user.

II. Computer as Modeling Device

A. *Games* — the use of the computer to model an interactive environment in which the user is required to outmaneuver, outthink, or outwit other users or the computer. Games challenge the user to reach a full enough understanding of their structure to master playing them. Thus, they may engage and help to develop a wide range of problem-solving skills. Though game elements are frequently used to enliven drill and practice or tutorials, pure games rarely teach directly.

B. *Simulations* — the use of the computer to model some aspect of reality or some set of real conditions so as to make the reality more amenable to manipulation and study. Simulations may cover a wide range of phenomena, from planetary motion to airplane flight to presidential elections to the battle of Gettysburg. They encourage the user to come to understand the rules that are at play in the model of reality that has been constructed. In theory, an understanding of the simplified model can lead to a better understanding of the more complex reality. Like pure games, simulations rarely teach anything in particular, but they encourage problem solving and frequently stimulate many other kinds of learning.

III. Computer as Tool

A. *Special Purpose Tool* — the use of the computer to carry out a specific, narrowly-defined task, usually a significant task that is frequently repeated. Tool programs of this kind are now beginning to proliferate because of their value in specific applications. For example, there are now numerous spelling programs which, working together with a word processor (a general purpose tool) will check any piece of writing for spelling mistakes.

B. *General Purpose Tool* — the use of the computer to assist people in carrying out a range of tasks within some general application area. Since the task is not specified ahead of time, it is up to the user to determine what is to be done and then to adapt the tool program to carry out that task. The power of general purpose tools is that the user may repeatedly adapt and readapt them in virtually limitless ways for a multitude of purposes, thereby creating a kit of special purpose tools. Several types of programs fall into this category: word processors, database managers, spread sheet programs, graphics utilities, music utilities, etc.

C. *Tool Making Tools* — the use of the computer to create new tools of either a special or general purpose. Some programs are so broadly general in their scope that they have no immediate use except to create tools. All computer languages serve this function. In addition, there are more and more programs becoming available that are designed specifically to help people design useful tools without having to learn a computer language or become a sophisticated programmer.

The Question Of Standards

With a set of descriptive categories such as the one presented above, it is then possible to develop a set of standards appropriate for each category of software. And each set of standards will differ substantially from the set that is appropriate for another category.

For the more prevalent kinds of educational software — drill and practice and tutorial — some preliminary standards have begun to develop (e.g., programs should not make judgments about user behavior). The recent vast improvement in video arcade games has helped to establish some preliminary standards for educational games (e.g., programs must respond quickly and accurately to user inputs). There are so few educational simulations and tool programs that we are just beginning to be able to consider what quality programs in these categories might be (e.g., it seems clear that a functional and efficient "user interface" is one of the critical features).

It is exceedingly important that any effort at software evaluation recognize that now is not the time for setting firm standards, but rather a time for developing flexible ones. We are at the very beginning of the application of computer technology to education. As noted before, software development is in its infancy, and so is hardware development. What was considered state-of-the-art a month ago is less than adequate next month. We are where the Wright brothers were at Kitty Hawk — just barely off the ground.

We must work to facilitate the emergence of excellence in software, not restrain it. Whatever standards we create must be flexible enough to embrace the next stages of discovery and innovation. The standards we set for any kind of software today are based on the software we experienced yesterday. They will certainly have to be adjusted to do justice to the software that is produced tomorrow.

The most effective way to keep in touch with the future is to maintain contact with groups working on software design and development. These groups, some in universities but most in private software companies, can be helpful in several ways:

1) They frequently write or give speeches about their work prior to publishing it, and these accounts explain the specifications they have set and the reasons for them.

2) They are sometimes willing to share internal documents or working papers that shed light on the standards they are setting for their work.

3) They are often willing to share pre-publication, test versions of their software for comments and reactions. Thus, there is an opportunity to sample new software well ahead of publication.

We must evolve a perspective on software evaluation that looks forward, not backward, that asks for continuing improvements, rather than preserving established practices, and that challenges the industry to give education the best hardware and software possible.

Evaluation Projects — A Sampling

Collected in this document are descriptions and some sample materials from a number of software evaluation projects. The following critical commentary of several of these projects is presented to clarify some of the serious dilemmas that are posed by trying to evaluate a new technology before that technology is well understood. I also wish to raise some questions about the appropriateness of the methodology used.

MicroSIFT

When MicroSIFT (Microcomputer Software and Information for Teachers) began operation in December, 1979, hopes were high that this federally-funded software clearinghouse, housed at the Northwest Regional Educational Laboratory, would bring organization and clarity to software evaluation. It was the first attempt of its kind. In its first year, MicroSIFT established a national network of cooperating software evaluation sites where teachers and other educators could carry out systematic evaluations. Evaluation criteria were carefully developed and tested. The criteria were put on forms, which were then thoroughly field tested. Finally, the *Evaluator's Guide* was developed to assist people in the evaluation process. In its second year, MicroSIFT published its first evaluations. Software chosen for evaluation represented a fairly narrow spectrum of the range of available software. Almost all of the evaluated programs used tutorial and/or drill and practice instructional techniques. In my opinion, the evaluations were ambiguous. But the differences between MicroSIFT evaluations and my own evaluations of some of the same software make an important point: no one can rely entirely on evaluations performed by someone else. At best, these can be clues which may assist us in making our own informed choices, but they must never be substitutes for our own responsibility to evaluate.

MicroSIFT, as the first attempt of its kind, broke new ground and illuminated problems yet to be solved: first, evaluation by committee may be a worthwhile process if the committee can hammer out a consensus. But if the committee never resolves differences and the consensus is created by nother party, the result may be a very odd stew.

Second. while the goal of MicroSIFT's methodology — to attain reliability and credibility by reducing the level of subjectivity in evaluation — is a worthy one, the result does not always make clear distinctions among different programs or differentiate good programs from bad ones. In spite of its wrinkles. MicroSIFT has provided educators with one of the first evaluation forms to attempt to collect data in some depth. In addition, the creation of a network of educators interested in evaluating software and sharing their evaluations was valuable in itself, and contributed to the growing body of knowledge about evaluation as a process.

EPIE

Since 1967. EPIE (Educational Products Information Exchange) had been conducting evaluations of educational materials, equipment and systems. In 1981. the EPIE Institute published one of the first major efforts to evaluate software (*Microcomputer Courseware/Microprocessor Games*). Thus, they brought to the task of evaluating this new technology many years of experience and a strong reputation in the field. In carrying out their software evaluations, they established a collaborative relationship with the Microcomputer Resource Center at Teachers College, Columbia University. Six programs are reviewed in EPIE's initial publication, and, like MicroSIFT, the selection is all from tutorial or drill and practice materials. Once again, the reader may be left with the incorrect impression that the selection represents what is available.

There is far less ambiguity in these reports. They reflect a stronger and surer point of view about what quality standards might be applied to educational software, at least within the limited range of programs they considered (a brief introduction outlines the software attributes they considered and why). While the reports are undoubtedly the product of a group effort, they reflect stronger consensus and therefore convey a better sense of what the software is and what it is not.

EPIE has recently joined forces with Consumers Union to publish both software (*Microcomputer Courseware PRO/FILES*) and hardware (*Microcomputer Hardware PRO/FILES*) evaluations in a new "filecard" format. They have also formed the Consortium for Quality in Educational Computing Products which will publish a regular newsletter (*MICROgram*) as a forum for consumer concerns and issues.

At this time, I have only one sample of their software evaluations to consider. Unfortunately, from this one sample, the standards that were set by EPIE's earlier efforts are not met in their new format or through their new methodology. The comments are briefer and made to sound more objective. A quantitative summary rating of major attributes is provided. There are brief quotes from magazine reviews and from student users, and there is one page of brief summary statements on various attributes of the program. While the format appears more accessible and there is less reading required, the overall impression is now more like the MicroSIFT evaluations, that is, ambiguous and uncertain. And, sad to say, the reader does not know after reading the evaluation whether the program is worth serious consideration.

Courseware Report Card

One of the most promising recent efforts to provide software evaluation is *Courseware Report Card*, published by Educational Insights of Compton, California. Interestingly, their reports have some of the good flavor of the early EPIE reports in that they present a reasonably coherent and consistent point of view.

The company claims that it is "the first publication to offer a large volume of detailed critical reviews of educational software for a variety of microcomputer systems." They are probably right. Their first volume of reports consists of evaluations of twenty-two elementary programs and twenty-two secondary programs (four programs are included for both levels). While most of the programs reviewed are tutorial or drill and practice, there are some games and a couple of simulations. The selection also includes programs published recently, suggesting that the authors are in touch with current software developments.

The evaluation format is refreshingly straightforward: a short introduction, a thorough description, an evaluation on six criteria (performance, ease of use, error handling, appropriateness, documentation, and educational value), a box of basic program information, and a box for a letter grade (A - F) on each of the six criteria.

Three positive features of these evaluations stand out. For the most part, when I have finished reading one of them, I know something about the value of the program. After reading several, I can distinguish one evaluation from another. And, over the entire group, real distinctions are made: some programs clearly emerge as very high quality, and others as low quality. For those that fall in between, the reasons are fairly clear.

Some Conclusions

Education lives with the belief that good evaluation must somehow be objective. Both MicroSIFT and EPIE have made strong efforts to objectify the software evaluation process through elaborate evaluation forms and systems for digesting evaluator responses. The result, as is often the case in educational evaluation, is a product that runs a risk of creating confusion. *Courseware Report Card* has no such process.

In a new field, where the greatest need is for everyone to become much more literate, it is probably not a good idea to have people looking to any sole source of authority. Rather, as I shall argue more fully below, the real need is for educators to become better informed so they can trust their own instinct and judgments. In our new computerized environment, we all face the responsibility of building our understanding through our own experiences, not through the processed experiences of others.

The evaluation conducted by both MicroSIFT and EPIE depend i : part on the completion, by trained evaluators, of extensive evaluation forms. Though the forms do allow for some unstructured responses, most responses are constrained by the questions posed on the form. On the one hand, the questions are intended to guide the evaluator and produce some consistency across evaluators. But on the other hand, they also force the evaluator to view the software in predetermined ways that do not always fit the software being considered. For example, having reviewed these forms, I feel they show a strong bias toward the evaluation of tutorial or drill and practice software. Many of the questions would not apply to a general purpose tool program, and the kinds of questions that would apply to this totally different type of software are not there.

There are some other problems with any centralized evaluation system. It is very slow at a time when advances in software development are being made very quickly. By the time a set of reports is published, most of the software that has been evaluated may have been improved or withdrawn from the market in favor of new and better products. It is no longer likely to be state-of-the-art. As techniques for software design improve, standards must also change. What was considered excellent a year ago may be only marginally acceptable today.

Most software does not stay the same for very long because improvements are usually easy to incorporate at any time. Software is constantly undergoing improvements, based upon user reactions and review in publications. Both MicroSIFT and EPIE intend their evaluations to be of help to publishers in improving the quality of their products. However, because of the inherent slowness of the evaluation process, it is possible that many of the criticisms made would be corrected by the time they reach the user.

As software develops and as users have more experience with a range of software types, we will come to understand more fully the vast potential of this technology for human improvement. Many serious thinkers have argued that computer use may well expand human learning beyond our current expectations, leading to a broadened and deepened human consciousness. We must be wary of software evaluation systems that have the effect of restraining our vision of what is possible rather than extending it.

In a fast-growing field, a reviewer has at least two major responsibilities: to maintain quality standards, judging software in relation to those standards, and to recognize and illuminate innovations that may well be the quality standards of tomorrow. It is hard to see how formal centralized evaluation systems can well serve either of these responsibilities.

Some Alternatives to Existing Methods

Probably the greatest need today is to facilitate the sharing of information about all of the various kinds of software and about programs that exemplify each kind. We ought to be using the capacities of the computer and of advanced communications technology to help us with this task. We can build a software database that would include as much software information as possible from a wide variety of sources. It could have several categories of information: general material on software and software evaluation, formal reports and software reviews, commentaries and critiques submitted by users, reactions and responses from publishers, etc. Some person or group could be assigned the task of setting up the database, maintaining it, and monitoring its use.

If well done, such a database would provide a valuable information resource in a form that would be easily accessible by anyone looking for information on a particular issue or a particular piece of software.

A database on microcomputer software, called RICE (Resources In Computer Education), has recently been established by the same group that produced MicroSIFT at the Northwest Regional Educational Laboratory. It is reported to provide information on about 2,000 educational software programs. About 200 of these will have been evaluated by the same network of cooperating institutions involved in the MicroSIFT evaluations. This database is currently accessible through the School Practices Information Network (SPIN), which is now being distributed by the Electronic Publishing Division of Scott, Foresman and Company.

I look forward in the very near future to the time when it will be possible for educators wanting to share information about computer software applications to communicate electronically with each other over a dedicated educational technology network. Such networks already exist in industry and higher education, and they provide users with an incredible resource for the sharing of ideas and information. Furthermore, the immediacy of the communication that is made possible from anywhere in the country to anywhere else, can have a special impact on solving complex problems confronting people in remote places.

Imagine, for example, that I'm a sixth grade science teacher doing a unit on the study of bones. I would like to know if educators on the network have either purchased or developed software themselves that might help me with my instruction. So, before I go home, I write a message for the network bulletin board and store it on a disk. Late in the evening, when the telecommunications rate are low, my message is placed on the network bulletin board. At various times the following day, people all over the country read my request, and a number of them send me responses. The following morning, I check my mailbox on the network and find to my delight that seven different commercial programs have been recommended for various aspects of teaching about bones, and two teachers have sent me programs they have written themselves, which I can store on my disk and try out later.

All the technology for such active sharing currently exists, and I think it will not be long before such sharing will be common practice.

One final point: good software takes advantage of all the features of computer technology — the interactive capacity, the speed, the graphic potential (including animation), the sound potential (including voice), and the dramatic potential. Because it is such a multi-dimensional medium, even the best evaluations of software fall considerably short of conveying to the potential user the nature of the software he or she wishes to know more about. With the advent of better telecommunication via computers, software publishers will probably offer demonstrations of the programs they are trying to sell, and software reviewers will be able to employ actual demonstrations of products in the course of their critical analyses. *WINDOW*, a new magazine in disk format, does exactly this type of software review. In time, this magazine may be available over the kind of educational network I have described here.

What Can Be Done Right Now?

All of this talk of future possibilities may not sound very helpful to people in schools who are faced with having to make software decisions right now. Where can they look for immediate guidance until some better system comes along?

Fortunately, there are a number of good sources of information on software. But, I would emphasize strongly that it is important always to take several sources of information into account in coming to any software decision. Probably the best sources are the educational computing magazines where software reviews for all types of computers are published regularly. Major computing trade magazines have been publishing special issues on education each year, and these are worth watching for because reviews of educational software are usually included (see Resource Information Section).

Unfortunately, the same software is not usually available for every type of computer, though there is a growing tendency for publishers to produce programs for several computers. There are many good computer-specific publications that review educational software. For example, *Peelings* is an Apple-specific publication that publishes nothing but software reviews, many of which are of educational programs. Apple's nonprofit Apple Education Foundation publishes *The Apple Journal of Courseware Review*, which includes extensive reviews of Apple software for education.

Perhaps there will soon appear educational software review publications for other computers, which will fill in some of the gaps in software information that currently exist.

There are two distinct advantages to taking the time to become immersed in the software review literature. Obviously, the information will lead to better software choices. But, more important, through sampling the ways in which various people approach the software revie process, one gradually develops a critical capacity of one's own. *Classroom Computer News* has emphasized this educational aspect of software reviews by providing a review tutorial to supplement the reviews in each issue. *InfoWorld* periodically publishes its guidelines for software and hardware reviews to inform the reader what it expects its reviewers to take into account in every review. Given the current stage in the growth of this technology, the strongest advice I can possibly give is for everyone to develop a critical capacity of his/her own.

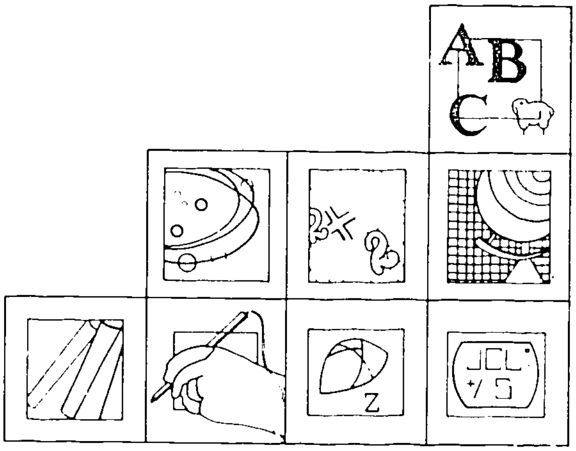
However, becoming conversant with the software review literature and developing a critical capacity is not enough. One must also spend time using software and attempting to apply, at least informally, one's critical capacity. I am constantly astounded by the number of educators I meet who seem to think that they can make intelligent educational decisions about computer software without taking the time to learn how to use the technology and become reasonably familiar with the software upon which the technology so completely depends.

There is one major problem in the software publishing industry that stands in the way of educators making informed choices of software: it is very difficult to get software on a preview basis or to return software that is found not to meet expectations. Yet this is a consumer's right that educators must insist on exercising, especially in a field where there is so much poor quality material. There is an encouraging trend now for publishers to change their position on this matter. I would urge all educators to help this trend by not purchasing from any company that does not permit them to evaluate software prior to purchase, or to return it after purchase (within a reasonable time period) if dissatisfied.

One further note of warning. It is never sufficient only to evaluate a demonstration program or a salesperson's quick overview of how a program works. The actual software being considered for purchase must be thoroughly tried under favorable conditions in order to be well evaluated.

I repeat that computer software, if it is any good at all, is not like a textbook. The use of the computer in education can be (and should be) substantially different from the use of the textbook or workbook. And, furthermore, the use of the computer is also substantially different from the use of other audiovisual aids. No other technology has such potential for our work as educators. So when we talk about the need for computer literacy in our schools, we must start by accepting that the first task is for us to become reasonably literate ourselves.

Then what's to be done? We have a professional responsibility to take every opportunity available to educate ourselves about computers and computer software so that we can make informed choices. It is not a responsibility that can be deferred to another time or referred to some outside agency. It is a critical responsibility that must be assumed right now by everyone involved in education so that our children will be able to function well in a changed and, I suspect, improved society, and so that our society can function well in a changed and, I hope, improved world.



Approaches To Software Evaluation

Several useful approaches to evaluating educational software have been developed during the last few years. In attempting to organize them, we have found that these approaches can be grouped along a continuum. At one end are those strategies designed to be used with large numbers of evaluators cooperating in an organized network under some given set of guidelines. At the other end of the continuumn are those approaches that provide discreet evaluation forms designed to be used freely by any individuals or groups at their own discretion.

In the first category, we would place MicroSIFT and EPIE/Consumers Union (EPIE/CU). Briefly, MicroSIFT's system is made up of evaluators at sites around the country who use MicroSIFT's evaluation form and the *Evaluator's Guide*. Results are summarized by each site's computing expert and these summary reviews are published by MicroSIFT; they are available from network members or through RICE, a public access database. (The *Evaluator's Guide* is now commercially available to educators who are not formal members of this network.) EPIE/CU has formed a consortium of school districts around the country to evaluate both software and hardware. Evaluation specialists at these sites, trained in using EPIE's evaluation form, supply EPIE/CU with their analyses, which are then published and available by subscription.

Next to MicroSIFT and EPIE/CU on our continuum, we would place *School Microware Reviews* and *Courseware Report Card*. While not formalized into a network, these publications do have access to evaluators, either in-house or out, who supply them with evaluations according to given criteria. These evaluations are then published and available by subscription. One notable difference in these approaches is that each provides publishers with the opportunity and the forum to respond to evaluations of their products.

Third along our continuum, we place Minnesota Educational Computer Consortium (MECC), SOFTSWAP, and CONDUIT. The primary function of these organizations is the development and/or distribution of software. Evaluation of their software is an essential ingredient to their respective processes, but evaluation is not the primary purpose of their existence. MECC and SOFTSWAP generally do not publish evaluations of their software. CONDUIT publishes evaluations in *Pipeline*, available by subscription.

At the far end of our continuum — in the category of discrete forms designed to be used freely by individuals or groups at their own discretion —are the evaluation forms developed by the National Council of Teachers of Mathematics (NCTM), Scholastic, Inc., and *Electronic Learning* magazine. Each of these forms has supporting material. The form from NCTM is part of a book called *Guidelines for Evaluating Computerized Instructional Materials*; the Scholastic, Inc. form is part of an inservice training kit; and the form published in *Electronic Learning* is described as a synthesis of several forms.

The information that follows includes an abstract and a copy of the evaluation form used in each approach. One minor note: excerpts from *Courseware Report Card* and *Guidelines for Evaluating Comptuerized Instructional Materials* (NCTM) make reference to other portions of these publications. The materials thus referenced will not be found here. Permission to reproduce the information in this section has been granted by each institution or publisher included.

Northwest
Regional
Educational
Laboratory

COURSEWARE EVALUATION PROCESS OVERVIEW

microSIFT

300 S.W. Sixth Avenue Portland, Oregon 97204 (503) 248-6800

The process described here was designed during the 1980-81 school year as a framework for the evaluation of microcomputer-based instructional materials by the MicroSIFT clearinghouse. The components are a set of forms, the Evaluator's Guide, and a network of educational institutions.

The forms were based originally on the forms developed and used by the CONDUIT Project for evaluating computer-based instructional packages for post-secondary institutions. They were modified with additional concepts adopted from forms developed by the organizations and individuals. The "Courseware Description" form identifies the factual information necessary for evaluation and use of a package, including source, ability level, subject, mode of instruction, required hardware and software, instructional objectives and prerequisites. The "Courseware Evaluation" form is designed to be used after the information on the Description form is available. A copy of the rating portion is on the reverse of this page. In addition, it provides space for identifying major strengths and weaknesses, and suggestions for potential classroom uses.

The Evaluator's Guide is a book designed to be used by teachers and others who are evaluating courseware for MicroSIFT. It describes the use of the Description and Evaluation forms, and provides guidelines, suggestions and interpretations of each item on the Evaluation form.

The microSIFT Network is a group of over 20 educational organizations serving elementary and secondary schools with computer services and other types of support. The network includes school districts, regional service centers, state departments and state consortia which have experience in serving local districts with inservice, software, computer time and services, curriculum materials and evaluation services. They have staff whose time is assigned to supporting the instructional computing activities of schools in their geographic area.

The components above are used in the three stages of the process described below:

1. Sifting - This is a first look at a package to determine that it is instructional in nature, will actually operate without problems on the appropriate microcomputer, and is complete with instructions. MicroSIFT staff complete this phase of the process.

2. Description - A package passing stage 1 successfully is described in this stage using the Description form discussed above. The producer and MicroSIFT staff complete this stage for the most part. However, some information may be supplied in stage 3.

3. Peer Review - Teachers with experience in the subject and grade or ability level of the material are selected from schools served by a network site to evaluate packages according to the Evaluation form and Evaluator's Guide. A package is identified for a network site by MicroSIFT staff, and the teachers are selected by the instructional computing expert at the site. After the evaluations are completed by the teachers, an evaluation is also done by the network site expert, who also completes a summary review encompassing all three evaluations. The summary review becomes the MicroSIFT evaluation of the package.

Completion of the first three stages takes approximately three months. The resulting evaluations are professional opinions based on experience, and are not necessarily based on observation of student use of the packages. While some do include such use, the evaluators are volunteers, and their time does not always allow for extensive student involvement. Also, a package may be evaluated at a point in the school year not in conjunction with the time the topic is studied.

A fourth stage of evaluation in greater depth is desirable for some packages because of their complexity or breadth of curriculum coverage. Such a stage might include pre- and post-testing, detailed observation of student activity while using a package, or other procedures. This stage is not being implemented by MicroSIFT at this time, although some approaches for it are being developed and investigated.

micro SIFT **COURSEWARE EVALUATION**

NORTHWEST REGIONAL EDUCATIONAL LABORATORY

Package title ______________________________ Producer ____________________

Evaluator name ____________________________ Organization __________________

Date ________________ □ Check this box if this evaluation is based partly on your observation of student use of this package

SA - Strongly Agree A - Agree D - Disagree SD - Strongly Disagree NA - Not applicable
Please include comments on individual items on the reverse page.

CONTENT CHARACTERISTICS

(1)	SA	A	D	SD	NA	The content is accurate.
(2)	SA	A	D	SD	NA	The content has educational value.
(3)	SA	A	D	SD	NA	The content is free of race, ethnic, sex and other stereotypes.

INSTRUCTIONAL CHARACTERISTICS

(4)	SA	A	D	SD	NA	The purpose of the package is well defined.
(5)	SA	A	D	SD	NA	The package achieves its defined purpose.
(6)	SA	A	D	SD	NA	Presentation of content is clear and logical.
(7)	SA	A	D	SD	NA	The level of difficulty is appropriate for the target audience.
(8)	SA	A	D	SD	NA	Graphics/color/sound are used for appropriate instructional reasons.
(9)	SA	A	D	SD	NA	Use of the package is motivational.
(10)	SA	A	D	SD	NA	The package effectively stimulates student creativity.
(11)	SA	A	D	SD	NA	Feedback on student responses is effectively employed.
(12)	SA	A	D	SD	NA	The learner controls the rate and sequence of presentation and review.
(13)	SA	A	D	SD	NA	Instruction is integrated with previous student experience.
(14)	SA	A	D	SD	NA	Learning can be generalized to an appropriate range of situations.

TECHNICAL CHARACTERISTICS

(15)	SA	A	D	SD	NA	The user support materials are comprehensive.
(16)	SA	A	D	SD	NA	The user support materials are effective.
(17)	SA	A	D	SD	NA	Information displays are effective.
(18)	SA	A	D	SD	NA	Intended users can easily and independently operate the program.
(19)	SA	A	D	SD	NA	Teachers can easily employ the package.
(20)	SA	A	D	SD	NA	The program appropriately uses relevant computer capabilities.
(21)	SA	A	D	SD	NA	The program is reliable in normal use.

QUALITY

Write a number from 1 (low) to 5 (high) which represents your judgement of the quality of the package in each division:

____ Content

____ Instructional Characteristics

____ Technical Characteristics

RECOMMENDATIONS

□ I highly recommend this package.

□ I would use or recommend use of this package with little or no change. (Note suggestions for effective use below.)

□ I would use or recommend use of this package only if certain changes were made. (Note changes under weaknesses or other comments.)

□ I would not use or recommend this package. (Note reasons under weaknesses.)

Describe the potential use of the package in classroom settings

Estimate the amount of time a student would need to work
with the package in order to achieve the objectives:
(Can be total time, time per day, time range or other indicator.)

Strengths:

Weaknesses:

Other comments:

microSIFT COURSEWARE DESCRIPTION

NORTHWEST REGIONAL
EDUCATIONAL LABORATORY

Title ______________________ Version Evaluated ______________

Producer ______________________ Cost ______________

Subject/Topics ______________________________

Grade Level(s) (circle) pre-1 1 2 3 4 5 6 7 8 9 10 11 12 post-secondary

Required Hardware ______________________________

Required Software ______________________________

Software protected? □ yes □ no Medium of Transfer: □ Tape Cassette □ ROM Cartridge □ 5" Flexible Disk □ 8" Flexible Disk

Back Up Policy ______________________________

Producer's field test data is available □ on request □ with package □ not available

INSTRUCTIONAL PURPOSES & TECHNIQUES
please check all applicable

□ Remediation
□ Standard instruction
□ Enrichment
□ Assessment
□ Instructional management
□ Authoring
□ Drill and practice
□ Tutorial
□ Information retrieval
□ Game
□ Simulation
□ Problem Solving
□ Other ______________

DOCUMENTATION AVAILABLE
circle P (program) S (supplementary material)

P S Suggested grade/ability level(s)
P S Instructional objectives
P S Prerequisite skills or activities
P S Sample program output
P S Program operating instructions
P S Pre-test
P S Post-test
P S Teacher's information
P S Resource/reference information
P S Student's instructions
P S Student worksheets
P S Textbook correlation
P S Follow-up activities
P S Other ______________

OBJECTIVES □ Stated □ Inferred

PREREQUISITES □ Stated □ Inferred

Describe package CONTENT AND STRUCTURE, including record keeping and reporting functions

[illegible] ack for more space

EPIE & Consumers Union

P.O. Box 620 Stony Brook, N.Y. 11790
Nonprofit • Consumer Supported • Unbiased

MICROCOMPUTER COURSEWARE EVALUATION FACT SHEET

The Educational Products Information Exchange (EPIE) Institute has, for many years, provided objective analyses of textbooks and other materials used in elementary and secondary schools. These analyses, used by numerous schools in the United States and Canada, have been an important part of the materials selection process. Now, with the advent of microcomputers in the classroom, it is essential that a process similar to that used in the analysis and selection of conventional classrcom materials also be applied to microcomputer courseware designed for classroom use.

With partial foundation support, the EPIE Institute and the Microcomputer Resource Center (MRC) at Teachers College, Columbia University began a project in 1981 that systematically analyzed selected microcomputer courseware designed for school use. For that project, the EPIE Institute's instructional materials analysis instrument was adapted to the analysis of microcomputer courseware. This instrument systematized the analysis of courseware in terms of its instructional design - developer's rationale, learner objectives, content, methods and approach, and tests and means of evaluation. In the two years since the development of that first courseware evaluation instrument,EPIE has continued to evolve what it believes is an approach to courseware evaluation that is being responsive to the changing and evolving field of educational computing.

In 1982, as the flow of microcomputer products being marketed for school and home use began to increase, EPIE,in addition to continuing its work with MRC, joined with Consumers Union of the U.S. (publisher of Consumer Reports) and the consortium of school districts, including Albuquerque, Boston, Cincinnati, Housotn, Detroit, and Salt Lake City, to evaluate both microcomputer courseware and hardware products. EPIE evaluation specialists have provided extensive training in using the EPIE courseware evaluation form to teams of evaluators in these cities. Once certified as courseware analysts, members of these teams supply EPIE and Consumers Union with analyses of the courseware considered for purchase by school districts. As EPIE's courseware instrumentation continues to evolve, these trained specialists will use these techniques.

Using the resources of EPIE, Consumers Union, MRC and the six school districts, it has been possible to analyze many of the major courseware packages and hardware systems and an increasing number of smaller courseware packages. These product evaluations are available on a subscription basis to all school consumers in the form of EPIE and Consumers Union Courseware and Hardware PRO/FILES (Product Files) and will be updated as new products are evaluated and new user evaluations are gathered on previously included products. Subscribers to EPIE-CU Micro-Courseware and Micro-Hardware PRO/FILES also become members of the EPIE-CU sponsored Consortium for Quality in Educational Computing Products and the International Council for Computing in Education (ICCE). ICCE publishes the consortium's newsletter, MICROGRAM, as part of its monthly publication THE COMPUTING TEACHER.

For further information about the consortium of school districts, the EPIE and Consumers Union Micro-Hardware and Micro-Courseware PRO/FILES and Evaluations, and other EPIE services, please call EPIE Institute at 212-678-3340 or 516-246-8664.

EPIE Institute Executive Director: P. Kenneth Komoski
Coordinator, Microcomputer Courseware Evaluation: Ellen R. Bialo
MRC Director: James Dunne

EPIE Institute

MICROCOMPUTER COURSEWARE EVALUATION FORM

Analyst's Name: ______________________ Date: __________

plete Program Title: ____

ducer: (name) ______________________ Copyright: ______________________

(address) ______________________ Author(s): ______________________

(phone) ______________________ ______________________

DWARE CONFIGURATION

rocomputer Model*	Minimum K	Version of DOS needed	Firmware and/or Peripherals needed

ease star that microcomputer model used for this evaluation

PROGRAM COMPONENTS PRICE*

(specify: number of disks, tapes, and/or ROM modules; teacher's guide w/number of pages; included support materials)

______________________ ______

______________________ ______

______________________ ______

*specify pricing if:

components available separately __________

reduced costs for backup disks __________

ended Users Specified by Producer:

ge and/or Grade Range ______________________

rouping (circle appropriate descriptor(s)) INDIVIDUALS PAIRS SMALL GROUPS CLASSROOM NETWORK

udent Entry Competencies (quote developer if possible, USE QUOTATION MARKS, infer if not specified)

____

iculum Role Specified by Producer: (circle appropriate descriptor(s) in each line)

THEMATICS SPELLING LANGUAGE ARTS READING SOCIAL STUDIES SCIENCE OTHER ______________ (specify)

PPLEMENTAL BASIC PROGRAMMING LANGUAGE (COMPUTER) MANAGEMENT OTHER ______________ (specify)

ILL and PRACTICE TUTORIAL EDUCATIONAL GAMING SIMULATION OTHER ______________ (specify)

CONTENT TOPICS: Stated or Inferred (circle one). If stated, USE QUOTATION MARKS (e.g., "Adding Fractions").

____

____

____

____

____

____

____

____

____

____

____

Cite page number(s): ____________

OTHER PROGRAMS ON DISK (if applicable)

____

____

GOALS AND OBJECTIVES

*Are goals and objectives supported by contents? YES NO

Describe ______________________

____

PROGRAM INTENTS

Developer's Rationale: Stated or Inferred (circle one). If stated, USE QUOTATION MARKS.

____

____

____

____

Cite page number(s): ____________

Development or Field Testing Evidence: Stated or Inferred (circle one). If stated, USE QUOTATION MARKS.

____

____

____

____

Cite page number(s): ____________

Learner Objectives: Stated or Inferred (circle one). If stated, USE QUOTATION MARKS.

3 Examples: ____________________

____

____

____

Cite page number(s): ____________

The rating scale (high, middle, low) is provided to assist you in making decisions about each question. Use the scale when appropriate, but always make remarks in the space provided, to clarify your response.

CONTENTS

* Describe what type of content is presented in what units (e.g., 10 games practice word attack skills at 3-5 difficulty levels). ______

______

	High	Middle	Low
APPROPRIATENESS FOR INTENDED USERS			
● Is the match of content to student ability levels appropriate?	1	2	3

Describe ______

______

● Is the program scope reasonable (given program length and activities)?	1	2	3

Describe ______

______

● Is the readability (vocabulary, sentence structure) appropriate?	1	2	3

Describe ______

______

● Is the tone of address appropriate?	1	2	3

Describe ______

______

● Describe uses of the program other than those suggested by the producer (e.g., special populations, grouping, individualization, etc.)

______

______

ACCURACY AND FAIRNESS

● Is the factual presentation accurate?	YES	NO

Cite inaccuracies ______

______

- Is the program free of errors in spelling, punctuation and grammar? YES NO

Cite inaccuracies ______________________________

____

- Is the content socially balanced? YES NO

Describe ______________________________

- Is the content free of stereotypes (racial, sex-role, ethnic). YES NO

Describe ______________________________

____

- Describe any potentially controversial content or methodology:

____

____

CLARITY

- Are directions clear? 1 2 3

Describe ______________________________

____

- Is display clear in terms of frame formatting and type readability? 1 2 3

Describe ______________________________

____

- Is display format consistent? 1 2 3

Describe ______________________________

____

- Is the use of examples and demonstrations effective? 1 2 3

Describe ______________________________

____

- Is the use of cues and prompts effective? 1 2 3

Describe ______________________________

____

METHODS AND APPROACH

TECHNICAL QUALITY

- Is the software free of programming errors? YES NO

Describe ____________________

____

- Are control keys used consistently? YES NO

Describe ____________________

____

- Is the program easy to run? YES NO

Describe ____________________

____

- Describe the terms of warranty:

____

____

APPROACH TO CONTENT

- Is the approach appropriate for intended users? 1 2 3

Describe ____________________

____

- Does the approach enhance the presentation of the content? 1 2 3

Describe ____________________

____

DOCUMENTATION (Manual, Teacher's Guide, etc.)

- Are there instructional suggestions? If so, describe their value: YES NO

____

____

- Does the teacher's guide assist in organizing and relating the other instructional components (workbooks, other materials) with the program? YES NO

Describe ____________________

____

METHODS AND APPROACH (continued)

- Does the teacher's guide provide for integrating the program with the basic curriculum? YES NO

Describe ____

____

- Are the technical explanations for implementation clear and complete? YES NO

Describe ____

____

- Does the producer recommend teacher training? YES NO

Describe ____

____

- If the teacher training is not recommended, in your opinion, is it necessary? YES NO

Of what should it consist? ___________________________

____

SUPPORT MATERIALS

- List and identify role: ______________________________

____

- Are support materials essential for implementation of program? YES NO

USER CONTROL

- Are there opportunities (including menus) for the user to choose among content topics (e.g., lessons, games, etc.)? YES NO

Describe ____

____

- Can the student review instructions? YES NO

Describe ____

____

- Can the student exit the program at any time? YES NO

Describe ____

____

METHODS AND APPROACH (continued)

- Can the student alter the rate of presentation (text rolling, problem display)? YES NO

Describe ______________________________

____

- Can the student call on Help or Hint-type options? YES NO

Describe ______________________________

____

- Can the teacher reset the parameters of the program? YES NO

Describe ______________________________

____

- Can the teacher add or change content? YES NO

Describe ______________________________

____

FEEDBACK

- Is feedback appropriate for the intended users? 1 2 3

Describe ______________________________

____

- Is feedback non-threatening? 1 2 3

Describe ______________________________

____

- Does feedback avoid reinforcing wrong response (as with an appealing graphic)? 1 2 3

Describe ______________________________

____

- Does feedback remediate? 1 2 3

Describe ______________________________

____

- Is feedback immediate? YES NO

Describe ______________________________

____

11/82

METHODS AND APPROACH (continued)

o Describe typical feedback after correct response: ____________________

____

o Describe typical feedback after incorrect response: ____________________

____

GRAPHICS

● Are graphics appropriate for the intended users? 1 2 3 NONE

Describe ____

____

● Are graphics a program feature? 1 2 3

Describe ____

____

● Are graphics embedded in content? 1 2 3

Describe ____

____

● Do graphics add interest? 1 2 3

Describe ____

____

● Do graphics avoid being distracting? 1 2 3

Describe ____

____

● Are graphics clear? 1 2 3

Describe ____

____

● Is there an option to use or not to use the graphics? YES NO

Describe ____

____

● Describe graphics (e.g., color, resolution, content, etc.): ____________

____

____

AUDIO

- Is there an audio component to the program? YES NO

Describe

- Does audio enhance the program? 1 2 3

Describe

- Can the audio be used with headsets and/or turned off? YES NO

Describe

RANDOM GENERATION

- Is random generation used in activities? YES NO

Describe

- Is random generation used in feedback? YES NO

Describe

IS THE PROGRAM EASY TO USE? 1 2 3

EVALUATION

*Does evaluation measure goals and objectives? YES NO

Describe

TESTS

- Does program provide for overall placement? YES NO N/A

Describe

- Are other tests included (lesson pretests, unit, mastery, printed)? YES NO N/A

Describe

EVALUATION (continued)

BRANCHING

- Does program automatically branch for review? YES NO N/A

Describe ______________________________

____

- Does program automatically branch after a lesson? YES NO N/A

Describe ______________________________

____

RECORDS MANAGEMENT

- Is there a record keeping system? YES NO

Describe ______________________________

____

- Is data stored for retrieval at any time? YES NO

Describe ______________________________

____

- Do student records identify specific difficulties? YES NO

Describe ______________________________

____

- Are student records cumulative? YES NO

Describe ______________________________

____

- How many students in how many classes do records hold? ______________

____

- Do class records identify specific difficulties? YES NO

Describe ______________________________

____

- Are class records cumulative? YES NO

Describe ______________________________

____

⊙ Is the management system easy to use? 1 2 3

Describe ____

____

⊙ Describe how program informs students of their progress (during/after lessons, after tests):

____

____

SAMPLE FRAMES

⊙ Describe the location of 3-5 representative frames: ______________

____

DESCRIPTION AND REVIEW

A. Describe the general structure of the program.

1. How many lessons in a unit? On a disk?
2. How are lessons structured?
   a. number of items
   b. appropximate working time

B. Describe how a student progresses through a typical lesson from beginning to end (e.g., entry, contingencies, instructions, menus, tests, etc.).

C. Describe the teacher's role in the use of the program (e.g., setting up a lesson, intervention, etc.).

D. Discuss the educational value of the program (e.g., importance of content, effectiveness of presentation, quality of instructional design).

E. Documentation

1. Describe the documentation (manual, teacher's guide, etc.).
2. Assess its usefulness technically.
3. Assess its usefulness pedagogically.

F. Management System

1. Describe the tests and record-keeping in the management system.
2. Assess the value of the tests.
3. Assess the value of record keeping system.

G. Assess the value of the support materials to the program.

SUMMARY

H. Discuss whether the program takes full advantage of the unique capabilities of the computer.

I. Describe the program strengths.

J. Describe the program weaknesses.

K. Make specific recommendations to the producer for revising and improving the program.

L. What is your considered judgment of the overall quality of the program?

CAPSULE RATING

Rate each of the following on a scale of 1 (lowest) to 10 (highest).

Overall rating of instructional design ________

Overall rating of software design ________

11/82

INSTRUCTIONS FOR PARTICIPATING IN THE SCHOOL MICROWARE USER SOFTWARE REVIEW PROGRAM

Here are the steps involved in having a product reviewed in SCHOOL MICROWARE REVIEWS:

1. The software supplier (you) selects a product and sends it to Dresden Associates, P.O. Box 246, Dresden, ME 04342. Preferably, this should be a fairly new product and one which you think is especially good.

2. Dresden Associates distributes software to evaluators based on their interests and capabilities. Evaluators sign an agreement not to copy the software made available.

3. The evaluator reviews the software, evaluates it using Dresden Associates' questionnaire, and returns software and questionnaire to Dresden Associates.

4. Dresden Associates sends the completed evaluation and a Courseware Description Form to the supplier (you) for comments and to collect purchasing information, respectively.

5. The supplier returns the comments and the completed Courseware Description Form to Dresden Associates.

6. Dresden Associates prepares a review using the evaluator's information and the information from the supplier. This is sent to both the evaluator and the supplier for final comment and approval.

7. The evaluator and supplier return the review draft and comments to Dresden Associates.

8. The review is published.

USER SOFTWARE REVIEW PROGRAM

As we indicated in the Introduction, we believe that improving software quality is essential to the success of educational computing and that the steps necessary for bringing about this improvement are well known. However, taking those steps will require enormous resources. Several agencies are working on the problem but have only scratched the surface.

This is where SCHOOL MICROWARE PUBLICATIONS and its readers come in. Because there are so many of us, we can accomplish a really big job with only a limited amount of individual effort and expense - and get a lot in return! We invite you to become a member of the growing network of educators all over the U.S. and Canada who evaluate software for SMW REVIEWS. Here is what's involved:

1. A prospective evaluator (you) obtains one or more educational programs. (We have programs for review if you don't - write or call as indicated below.)

2. You evaluate the software using either the form provided here (copy as necessary) or the Courseware Description and Courseware Evaluation forms used by the MicroSIFT Project. (See page 1 for a description of MicroSIFT; their forms and supporting documentation are available for $2.00 from Dresden Associates.) You return the completed evaluation forms to Dresden Associates.

3. Dresden Associates prepares a review from your evaluation and sends a draft copy to you and to the software producer for comments. (We of course reserve the right to accept or reject individual evaluations, based on our assessment of their merits and the criteria stated below.)

4. You and the producer return the review draft and the comments to Dresden Associates.

5. Dresden Associates incorporates the comments in the review and publishes it.

6. You receive copies of SMW REVIEWS. To compensate for the additional effort required, you get more copies if you evaluate large, multi-program products.

The bottom line is that SWM readers can get inexpensive access to essential, in-depth evaluative information.

ON SELECTING SOFTWARE FOR REVIEW

First, we would prefer that programs be for the Apple, Atari, PET, TI-99/4, or the TRS-80 Model I/III. Second, a product selected for review should be either an independent, free-standing program or an entire system of integrated programs. However, it should not be only one member of an operationally integrated set.

For example, several companies offer disk-oriented sets of programs to teach all aspects of a given broad topic, e.g., decimals. Programs may be included to pre-test; to tutor several sub-topics such as addition, conversion, and percentage; to record and report performance; and to coordinate the operation of the entire set. These programs are integrated, not only by topic, but also in an operational sense. That is, it is difficult or impossible to run one of them without having others available in the computer at the same time. It would be desirable to review several programs of the set (on one or more forms), but not just a single program, as the latter does not give a complete and accurate picture of the entire product.

On the other hand, most suppliers offer multiple-program packages, which must be purchased as a single unit, but which consist of several programs which can and do operate independently, i.e., not more than one of them needs to be available for any given one to run. Each of these programs is a suitable candidate for evaluation, assuming that it meets the other criteria stated here.

You have a better chance of getting your evaluation published if the product is not one that is likely to be reviewed for this project by several other persons. Evaluating a program two or three times is OK - in fact is preferred - but any more than that is probably a waste of time. We are not likely to publish more than two reviews of the same product unless an additional one offers some very unusual and useful insights. Please contact us by mail or phone if you need guidance with regard to selecting a product to evaluate.

THE EVALUATION FORM

Our form has been used successfully by many persons but has not been validated. Therefore, it is not offered as a finished product but as one which we can improve as we go along. Please feel free to make suggestions.

Instructions for use of the form are included in it. Please take the time to accurately describe the product in the space for Functional Description, and be generous with comments to support the numerical ratings you have assigned. If in doubt, take a look at some reviews done by other evaluators. We have found that it takes one to two hours to do a complete evaluation of a single program.

SUBMITTING EVALUATONS

Completed evaluation forms and other correspondence regarding review software should be addressed to Barbara Courter, Research Coordinator, Dresden Associates, P.O. Box 246, Dresden, ME 04342 (Tel: 207/737-4466).

SCHOOL MICROWARE EVALUATION FORM

(COPY AS NEEDED)

Your Name________________________Organization______________________________Position____________

Address____Tel:________________________

Product Name____________________Supplier______________________Price $__________No. of Progs. Under This Name_____

Subjects and Grades to Which Applicable____Est. Student Time Required____________

FUNCTIONAL DESCRIPTION - Describe the program in terms of its goals and what it does to achieve them. Give as much detail as possible (use extra sheets as necessary).

____

____

____

PRELIMINARY CONSIDERATION - Does this program contribute to the teaching of topic(s) which should be taught in today's schools? ___Yes ___No If No, give your reasons for this answer in the Comments section at the end of the form and omit the balance of the questionnaire.

INSTRUCTIONS - Enter a number in the blank to indicate the extent to which the product fulfills the description in the item, as follows: 2 - Completely, 1 - Partially, 0 - Not at All. If the item is not applicable to the product, enter N/A. If the item is unclear, enter U. Elaborate on answers as necessary in Comments at end or on extra sheets, giving item numbers.

DOCUMENTATION - List materials accompanying the program, e.g., teachers guide, student workbook.

1. Indicate types of information included.

----- a. Suggested course/subject, grade levels.

----- b. Goals.

----- c. Performance objectives.

----- d. Suggested teaching strateg(ies).

----- e. Correlation with standard texts.

----- f. Prerequisites for use of program.

----- g. Student exercises, teacher answers.

----- h. Operating instructions.

----- i. Listing and sample runs of program(s).

----- j. If a simulation, description of the model used.

----- k. Suggested topics for follow-up discussions.

----- l. Suggested references/activities foı follow-up.

----- 2. The documentation is written clearly.

----- 3. If a workbook is included, the format and content are appropriate.

INSTRUCTIONS GIVEN TO USER BY PROGRAM

1. The instructions are adequate regarding:

----- a. The instructional task to be performed.

----- b. Details of how to interact with the program.

----- 2. User has the option of skipping instructions if already known.

STUDENT-COMPUTER DIALOG

----- 1. Output is displayed screen by screen (paged) rather than scrolled.

2. If output is paged:

----- a. User has control over continuing to the next page.

----- b. Amount of information in each page is appropriate.

----- c. The perceptual impact (amount of type and lines) is suitable.

----- 3. Output is spaced and formatted so as to be easily readable.

----- 4. Language is well suited to most students' reading ability.

----- 5. Uses correct grammar, spelling, hyphenation and punctuation.

----- 6. Any grid or coordinate system used is consistent with common conventions.

----- 7. Students can respond with common symbols & ways of using them, e.g., right to left entry of sums.

----- 8. Accepts abbreviations for common responses.

----- 9. Provides for individual needs, e.g., opportunity to work with harder or easier material.

----- 10. Dialog is personalized, i.e., makes appropriate use of student names.

----- 11. Uses devices to get & maintain interest, e.g., variation of computer responses, humor, pace change, surprise.

12. Makes good use of any special features computer:

----- a. Graphics ----- b. Color ----- c. Sound

----- 13. Reinforcing responses (indications of right, wrong, etc.) are appropriate.

----- 14. The number of wrong answers allowed is reasonable.

----- 15. Responds appropriately if allowed number of wrong answers is exceeded.

----- 16. Provides opportunity to get help if difficulty is encountered.

----- 17. Minimizes bad entries via devices such as objective formats (multiple choice, etc).

----- 18. Deals well with inappropriate entries, i.e., response to typing errors, etc., is intelligible and useful.

----- 19. Required entries are within students' capabilities (esp. typing, vocabulary).

----- 20. Reports student performance periodically and at end of session.

MISCELLANEOUS CONCERNS

----- 1. If a simulation, the program gives a sufficiently accurate representation of the situation simulated.

----- 2. The concepts and vocabulary required to use the program are reasonable.

----- 3. Operates properly and is free of bugs.

----- 4. Is well structured and documented internally to facilitate any necessary debugging/modification.

COMMENTS - Please use this space and additional sheets as necessary to provide any information which you believe would help someone who is thinking about buying of the product being reviewed. In particular, indicate what you like best and least about the program. Also, list any changes which should be made.

____

____

____

Revised 8/82

COURSEWARE report card ELEMENTARY

Price: $12.50

Evaluations of Microcomputer Programs for Education

Volume I September 1982 No. 1

At last it's here!—the premier issue of COURSEWARE REPORT CARD, a software evaluation service for elementary and secondary educators.

We initiated COURSEWARE REPORT CARD after having become aware of the serious problems that exist concerning the selection of educational microcomputer software. There is a tremendous volume of software to choose from with more programs appearing every month; for various reasons very little of this material is available for previewing, and unfortunately a great deal of the product is not up to the standards teachers expect from more conventional educational media.

As for published evaluations, we found that many publications feature a few software reviews per issue, others carry reviews in greater quantity but on a relatively superficial and uncritical basis, while still others publish in-depth, analytical reviews in quantity but are connected with one or another of the major computer manufacturers and review only programs for that particular make of computer. To our knowledge COURSEWARE REPORT CARD is the first publication to offer a large volume of detailed, critical reviews of educational software for a variety of microcomputer systems.

Selection of Software for Review

COURSEWARE REPORT CARD is reviewing programs for Apple, Atari, PET/CBM, and Radio Shack TRS-80 microcomputers. This spring we examined the most recent catalogs of some 55 software producers and requested one or more programs from each for review. Most of the programs reviewed in this issue were sent to us in response to these requests. Others were received unsolicited from publishers or were made available to us by teachers, software dealers, or educational media centers. In our selection, we attempted to achieve a balance with regard to subject areas, grade levels, and hardware compatibility.

With the exception of two teaching-aid programs, all the software reviewed in this issue is in the realm of computer-assisted instruction (CAI). Future issues will also include reviews of management programs, authoring systems, and computer-literacy programs, including a survey of available versions of the LOGO and PILOT languages.

Each COURSEWARE REPORT CARD review is self-contained and three-hole punched for notebook storage by subject area, grade level, or hardware compatibility. A complete list of programs reviewed in this issue can be found on the back of this package.

Format for the Reviews

Each review consists of an abstract briefly summarizing the nature and scope of the program, an overall view of the program and its major features followed by a detailed description, and an evaluative section in which the program is examined critically in six areas: Performance, Ease of Use, Error Handling, Appropriateness, Documentation, and Educational Value. These categories are not absolute, and there is a great deal of overlap among them. In general, Performance refers to the program's overall content: electronic, verbal, and graphic. Ease of Use is concerned with use by both students and teachers relative to grade level and content. Error Handling addresses the question of how the program reacts to input errors. Appropriateness refers to the ways in which the program makes use of the capabilities of the computer. Documentation covers the print-media instructional and support materials, and Educational Value is self-explanatory (in management and teaching-aid programs, the heading Usefulness is substituted for this category).

The capsule summary on the first page of each review that rates the program from "A" to "F" in each of these categories should be balanced against the more detailed evaluative material. In many cases a reviewer might have based a letter grade on criteria that a particular teacher might not find relevant to his or her needs. Views expressed in COURSEWARE REPORT CARD are to a great extent subjective, and as educators our readers need not be reminded of the arbitrariness of grades. In general, each reviewer has full discretion in assigning grades, with one exception being in the

COURSEWARE REPORT CARD/Elementary is published five times per year by Educational Insights, Inc., 150 West Carob Street, Compton, CA 90220; phone (213) 637-2131 or (213) 979-1955. Subscription rates (one year): $49.50; for both Elementary and Secondary editions, $95. Canadian subscribers add $7 U.S. for each edition. Single copies $12.50; Canada, $14.50. Each article and entire contents copyright © 1982 by Educational Insights. Printed in the United States of America. The making of any copies or transcriptions of any of this material without the written permission of Educational Insights is prohibited. Requests for special permission or for multiple copies of individual reviews should be sent to the publisher.

COURSEWARE report card

Publisher
Burton Cutler

Managing Editor
Mark Falstein

Consulting Editor
Terry Garnholz

Senior Editor
Emily M. Hutchinson

Staff Editors
Laura Cohen,
La Vonne Miller-Casey, Marcia Shank

Contributing Editors
Linda Marie Hary, Terry Humphries,
Laurence Johnson

Editorial Assistant
Rita Hoferer

Art Director
Phillip Garcia

Production Assistants
Patricia Collins, Steven Rowley

Circulation Manager
Diana Cutler

Appropriateness category. It is our contention that drill and practice is not a particularly appropriate use of computers in education, and it is our policy not to assign any drill-and-practice program a grade higher than a "C" in this category unless it is enhanced by additional features.

About Our Reviewers

Any unsigned review appearing in COURSEWARE REPORT CARD was prepared by one or more members of our editorial staff: Mark Falstein, Laura Cohen, Terry Garnholz, Emily M. Hutchinson, La Vonne Miller-Casey, and Marcia Shank. COURSEWARE REPORT CARD is a publication of Educational Insights, a long-established publisher of supplementary educational materials. Every editor on the staff is a former classroom teacher and has had extensive experience in the evaluation, development, and production of curriculum materials for a wide variety of subject areas, grade levels, and applications. Managing Editor Mark Falstein is the author of *The Computer Is Here,* a computer-literacy activity kit for children and has served as editor of *The Electronic Classroom,* a newsletter on computer applications in education.

The following editors also contributed material to this issue of COURSEWARE REPORT CARD (Secondary edition only):

Linda Marie Hary chairs the mathematics department at John Glenn High School, Norwalk, California, and is a member of the Computer Instruction Committee of the Norwalk-La Mirada School District.

Terry Humphries directed the implementation of microcomputer-based education in the Bellflower Unified School District, Bellflower, California. He has taught programming and computer literacy to administrators, teachers, parents, and students; and served as a computer consultant to other school districts in southern California.

Laurence Johnson, a math teacher at South Pasadena (California) High School, has used computers extensively in his classes for the past three years and is currently working under a district grant to develop a computer-based mathematics curriculum.

Any reader who feels that he or she is qualified to become a contributor to COURSEWARE REPORT CARD is encouraged to send us a résumé and an appropriate writing sample. We're particularly in need of reviewers of software for the PET/CBM and TRS-80 systems. We will pay [illegible]0 for each review we accept for publication. Unsolicited manuscripts are not encouraged, but will be considered if written according to our format and specifications and accompanied by a self-addressed stamped envelope.

Acknowledgments

We'd like to thank the following individuals and institutions who provided us with access to hardware and/or software, consultation, advice, or encouragement: most especially Zhita Elvord and Marjorie Masters of the Professional Reference Center of the Los Angeles County Board of Education; also Jeff Kwiecien, manager of the Radio Shack Computer Center of Lakewood, California; Kevin Radke, manager for Computer Education Sales at Opportunities for Learning, Chatsworth, California; Bill Russell, President of AMIS Computers, Inc., Los Alamitos, California; Ann Lathrop, director of SOFTSWAP, the software exchange service of the San Mateo Educational Resource Center, Redwood City, California; and private citizens Noah Falstein and Derek Garnholz.

Complaint Department

Any software publisher who feels we didn't give his or her product a fair evaluation is invited to submit a letter of reply. We will publish all such letters that are limited to 200 words in length.

Teachers and administrators who have used programs reviewed in COURSEWARE REPORT CARD and who would like to offer corroborating or dissenting opinions are also encouraged to write to us. Please limit your replies to 100 words.

We'd like to know what the educational computing community in general (producers as well as consumers) thinks about our new publication. Write, or if you're attending EdCom in Los Angeles October 21-24, stop by our booth. We'll be the one flanked by the four big bodyguards trained in kung-fu.

MINNESOTA EDUCATIONAL COMPUTING CONSORTIUM

MECC

WHAT IS MECC?

The Minnesota Educational Computing Consortium is a public organization established by the State government to assist Minnesota schools and colleges implement educational computing. MECC was established in 1973 and began providing direct services to educators in 1974. Three types of services support the use of computers in instruction: 1) providing low-cost access to computing resources including the operation of a statewide timeshare system and the establishment of microcomputer purchase contracts; 2) development of educational courseware; and 3) inservice training of educators. MECC's knowledge and expertise in the educational computing field comes from nearly a decade of working with and providing leadership for hundreds of local educators on a daily basis.

Contact MECC at 2520 Broadway Drive, St. Paul, MN 55113-5199, 612/638-0600.

Note about the following forms: "Computer Courseware Review Form" is most useful for discussion and comparison of software, perhaps by a committee. It may be used as a means of discarding obviously poor software. The "Microcomputer Educational Materials Evaluation" and "Student Evaluation" are more than checklists; they may be used by teachers actually using software in the classroom, and provide room for extensive notation.

MINNESOTA EDUCATIONAL COMPUTING CONSORTIUM

Instructional Systems Development
2520 Broadway Drive
St. Paul, Minnesota 55113

MICROCOMPUTER EDUCATIONAL MATERIALS EVALUATION

Reviewer's Name ______________________ Review Period From: __________
To: __________

School/District ______________________

MECC will use your classroom evaluation of the enclosed educational materials as a basis for a review to be published in the USERS newsletter. In your evaluation, consider the total package—software and support material.

Title of Package: ______________________

List the programs from the package that you trialed with the students: ______________________

Classroom Review Situation

Please descibe the number and ability levels of students who used the program. Describe the situation in which it was trialed.

I. *Ease Of Use*

Can students operate the program easily? Are directions clear? Is it easy for students to start the programs?...end the programs? Is requested input easily understood?

II. *Level of Student Interest*

Did the students who tried the programs finish without your prompting? Did you feel that they were interested in and motivated by the materials? Did students want to do the activities again?

III. *Support of Teaching Process*

Was the material easily integrated with classwork? Did you feel it was a valuable instructional tool for communicating its stated objectives?

IV. *Use of Microcomputer Capabilities*

Was the interactivity of the microcomputer used to its full extent: Was the student addressed by name? Was feedback on student responses immediate and specific? Did students have control over rate or level or sequencing of presentation? Did students receive information on how well they were doing?

Were graphics, sound, color used effectively?

V. *Documentation*

Was the support documentation clear?...complete?...useful? Did it identify prerequisite learnings? Was it free of stereotyping?

VI. *Technical Accuracy*

Did you have any technical problems with the program?

VII. *Educational Content or Value*

Was the material accurate?...important?...educationally valuable?...appropriate for the audience? Was the quality of instruction generally good?

Summary

This program is educationally sound, and I would recommend it to other schools.

strongly disagree _____ : _____ : _____ strongly agree

Summary Comments

M.E.C.C.
STUDENT EVALUATION OF MICROCOMPUTER MATERIALS

STUDENT'S NAME ______________________________ GRADE __________

SCHOOL ______________________ TEACHER ______________________

DATE __________________ PACKAGE TITLE ______________________

PROGRAM NAME ______________________ APPROXIMATE AMOUNT OF
TIME SPENT ______________________

CIRCLE THE BEST ANSWER:
(Write comments if you want)

Using this lesson was: Comments: ______________________

1. Easy
2. Somewhat hard
3. Very hard

After doing this lesson I would: Comments: ______________________

1. Like to do another like this
2. Rather not do any more lessons like this

From doing this lesson I learned: Comments: ______________________

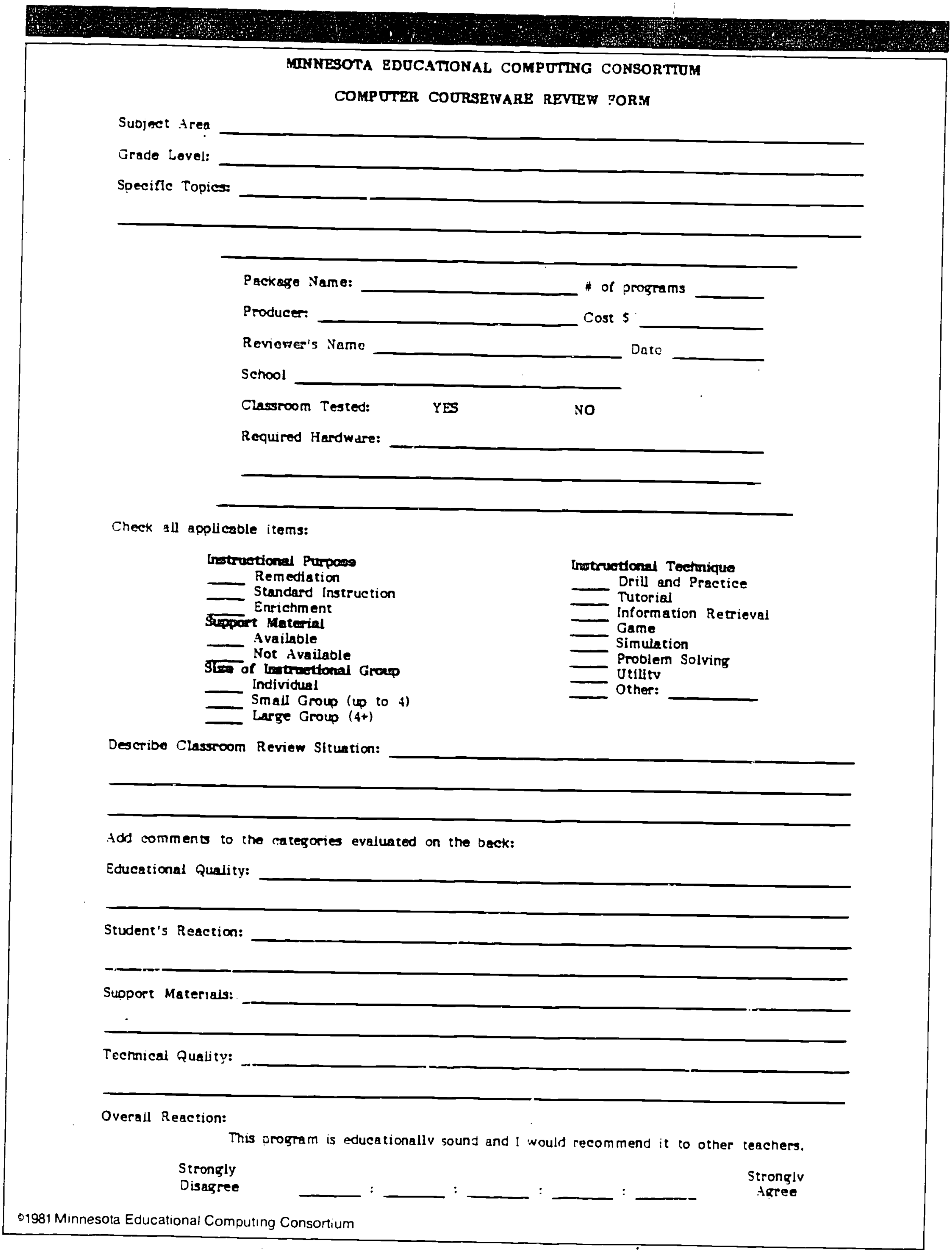
1. A lot about the subject
2. A little bit about the subject
3. Nothing

Describe what you would do to make this lesson better. ______________________

____

____

Thank you for your help!



MINNESOTA EDUCATIONAL COMPUTING CONSORTIUM

COMPUTER COURSEWARE REVIEW 7ORM

Subject Area ____________________

Grade Level: ____________________

Specific Topics: ____________________

____

____

Package Name: ____________ # of programs ______

Producer: ____________ Cost $ ______

Reviewer's Name ____________ Date ______

School ____________

Classroom Tested: YES NO

Required Hardware: ____________________

____

____

Check all applicable items:

Instructional Purpose
____ Remediation
____ Standard Instruction
____ Enrichment

Support Material
____ Available
____ Not Available

Size of Instructional Group
____ Individual
____ Small Group (up to 4)
____ Large Group (4+)

Instructional Technique
____ Drill and Practice
____ Tutorial
____ Information Retrieval
____ Game
____ Simulation
____ Problem Solving
____ Utility
____ Other: ______

Describe Classroom Review Situation: ____________________

____

____

Add comments to the categories evaluated on the back:

Educational Quality: ____________________

____

Student's Reaction: ____________________

____

Support Materials: ____________________

____

Technical Quality: ____________________

____

Overall Reaction:

This program is educationally sound and I would recommend it to other teachers.

Strongly Disagree ______ : ______ : ______ : ______ : ______ Strongly Agree

COURSEWARE EVALUATION FORM - PAGE 2

EDUCATIONAL CONSIDERATIONS

Do the Instructional Objectives match those of the district curriculum projects?	YES	NO
Are the objectives clear?	YES	NO
Are the concepts presented logical?	YES	NO
Is the information accurate?	YES	NO
Is it of appropriate difficulty level?	YES	NO
Is it sound theoretically (based on Learning Theory)?	YES	NO
Is is appropriate use of the computer?	YES	NO
Has it been field tested?	YES	NO
Is it at the appropriate reading level?	YES	NO
Can instructors adapt package to their objectives?	YES	NO

CLASSROOM

Are objectives defined to student?	YES	NO
Is it easy to use?	YES	NO
Are the directions clear?	YES	NO
Is the feedback effective?	YES	NO
Are the displays effective?	YES	NO
Are students provided information on how well they are doing?	YES	NO
Is it easy to implement?	YES	NO

SUPPORT MATERIAL

Are instructional objectives defined?	YES	NO
Is it necessary to use the program?	YES	NO
Does it give teacher background information necessary to operate?	YES	NO
Does it provide student material?	YES	NO
Does it tell teacher how to incorporate in instruction?	YES	NO

TECHNICAL QUALITIES

Is the text grammatically correct?	YES	NO
Is the text or the screen readable, (uncluttered, without scrolling)?	YES	NO
Is adequate time given to read the text?	YES	NO
Are graphics, color, and sound used effectively?	YES	NO
Are the questions clear?	YES	NO
Is input analyzed effectively?	YES	NO

ADDITIONAL QUALITIES - (Check all that apply)

____ **Drill and Practice/Game**

___ program provides difficulty levels

___ additional practice provided on problems lacking mastery

___ record of student performance

___ user in control of problems

___ feedback

 ___ positive

 ___ negative

 ___ provides hints

 ___ tutorial

____ **Tutorial**

___ evaluation included in learning

___ student is an active learner

___ sequence appropriate

___ segments can be accessed by learner

___ feedback

 ___ student on wrong response

 ___ provides clues

 ___ reinforces learning

____ **Problem Solving/Simulations**

___ segments accessed out of sequence

___ provides hints, clues

___ enables students to generalize

___ provides graphic representation

___ identifies assumptions (if simulation)

___ model valid

____ **Utility/Information Retrieval**

___ allows for maximum flexibility

___ low degree of complexity for user

___ provides necessary options to achieve objectives

How to Start a Software Exchange

by Ann Lathrop and Bobby Goodson

"What is different about SOFTSWAP? Isn't it just another users' group?"

Visitors to Softswap receive assistance from Microcomputer Center workers and from one another.

The Microcomputer Center in Redwood City, California, is a gathering place for members of Computer-Using Educators (CUE) and visitors from other areas who enjoy sharing their ideas, problems and expertise in this unique resource center. It's an exciting place to spend an afternoon, a few days, or longer, and the welcome mat is always out for educators who want to help with various CUE projects, to copy SOFTSWAP programs, or who simply like to sit and talk with other microcomputer enthusiasts who happen to drop in. Questions and answers fly back and forth, and frequent arguments as to which answer is correct reflect the diversity of opinion and experience represented by the group. At any one time the questions may vary from "What computer is best?" and "Can I do the school budget and attendance on this one?" to "How can I get this program to run off of our Corvus?" and "What is the difference between 3.2 and 3.3 DOS?"

Many answers are derived from do-it-yourself experience with the microcomputer systems, commercial packages, and SOFTSWAP programs which are available to visitors. Other answers are found by consulting the resource file of names and addresses of people experienced in almost all aspects of microcomputing. The books, journals and extensive reference files in the library provide still more answers to visitors' questions. The people in the Center, both staff and visitors, are also a valuable source of information and expertise.

The Microcomputer Center was established just over a year ago in the Library of the San Mateo Educational Resources Center (SMERC) as a joint project of Computer-Using Educators and the San Mateo County Office of Education. Visitors have come from throughout California, from many other states and Canadian provinces, and from Australia and France. All have been interested in learning more about the challenging potential of microcomputers as new instructional media. By supporting the establishment of this unusual educational resource, the San Mateo County Office of Education, under the leadership of Superintendent William K. Jennings, continues its traditional role of developing innovative educational programs in California.

There has been no outside funding of the Microcomputer Center. The San Mateo County Office of Education provides the space, maintenance and part-time staff support as a service to educators in the county and throughout the San Francisco Bay area. The active involvement of Computer-Using Educators as co-sponsor of the Center has been the other major factor in its success.

CUE has grown into an organization of over 1600 educators, with members in 40 states, 12 provinces and 12 foreign countries. CUE members donate hundreds of hours of volunteer time to organize and participate in activities of the center, including work on SOFTSWAP programs, demonstrations of new equipment and software, and the commercial software evaluation project.

Many members of the county office staff cooperated in the development of the Microcomputer Center and continue to be active in its operation. The original concept was presented by Ann Lathrop, Library Coordinator, who continues to supervise the SOFTSWAP and the software evaluation project. Assistance and support for the project was given by Dr. Curtis May, Director of Library Services, whose commitment has been essential to the successful establishment of the center. Visitors receive friendly help from Janice Marshall, Library Assistant, and from her staff of student aides. Curricu-

Recipe for a SOFTSWAP

Take one enthusiastic, experienced group of COMPUTER-USING EDUCATORS,

Add one COUNTY OFFICE OF EDUCATION with an established reputation for innovative leadership,

Stir in a collection of PUBLIC DOMAIN EDUCATIONAL SOFTWARE contributed by CUE members,

Season with generous loans of MICROCOMPUTERS from Commodore, Radio Shack, APPLE, CompuColor, and Bell & Howell,

Combine all ingredients with one LIBRARIAN who has strong organizational skills and a firm commitment to educational uses of microcomputers,

Sprinkle this rich mixture with many hours of CUE MEMBERS' VOLUNTEER WORK,

Let marinate for about ONE YEAR.

Recipe yields one SOFTSWAP . . . freely and gladly shared with Computer-Using Educators everywhere!

Ann Lathrop is Library Coordinator at the San Mateo County Office of Education, Redwood City, CA.

Bobby Goodson is President of Computer-Using Educators, and Computer Resource Specialist at the Cupertino Union School District, Cupertino, CA.

Reprinted by permission from Compute Publications, Feb., 1983. Small System Services, Inc., September. 1981. P.O. Box 5406, Greensboro, N.C., 27403, 919/275-9809. 12 issue sub. $20 U.S.

tributed software could be reduced or eliminated if donors would check the following:

1) Does the program run with relatively few problems? (Unfortunately, we receive many programs which do not load/run at all.)
2) Does the program trap input errors? Does it crash if the user inputs numeric responses when the computer is expecting alpha, or vice versa? What happens if the user hits an unwanted ENTER/RETURN? When there is an input error, is the computer response clear and simple? "Input error, redo from the start" is not helpful to the novice.
3) Are all the words spelled correctly?
4) Are the instructions and other information easy to read on the screen? Is it formatted to avoid split words at the end of a line? It is best to avoid crowding and to clear the screen at appropriate intervals. Instructions should be written at a reading level appropriate for the intended user.
5) Is the *positive* reinforcement for the correct responses more interesting that the negative reinforcement for errors? Graphics in most "Hangman" type games, for example, are usually more exciting when the user loses and is "hanged." Students will often deliberately miss a problem or a word in order to see a rocket explode, a boat sink, etc. It is best to try to provide strong encouragement for doing a task correctly and yet avoid over-praising. It is also better to provide several responses for both correct and incorrect answers – a little variety adds a great deal to the program.
6) Have you included REM statements as needed? They make it easier to use a program effectively. It is especially helpful to include statements telling the teacher how to modify word lists, problems or other entries.

Bobby Goodson, one of the moving forces of Computer-Using Educators, possesses a wealth of information about educational software.

7) Do you make negative remarks to or about the student? Sarcastic statements can be discouraging. We remove comments like "Boy was that dumb," "Stupid," "Dummy," etc.
8) Do you provide hints and/or eventually give the user the correct answer? We usually allow only 3 or 4 incorrect responses before providing either a hint or the correct response, often with some instruction of explanation. Exceptions are programs which are tests or timed drills for accuracy.
9) Is the content accurate? Are the definitions and/or explanations simple and straightforward? Do they correspond to the material in the more frequently used texts?
10) If it is a drill-and-practice program, do examples come up in random order? Are there enough problems or questions that the student will not have all the same examples if they run the program again? If it is appropriate, does the student have the option to return to the beginning of the exercise and try it again?

Finally, we want to be sure that the program submitted is truly an original creation of the author. If it is an adaptation from another program, from a book or a journal we want to be aware of that fact. We will request the author's permission to include the program in our collection. If we discover that a SOFTSWAP program is actually an infringement on someone else's copyright, we remove it from the dissemination disk.

The most exciting programs we receive are the creative ones that do more than just put workbook pages and drill problems on the screen. Contributed software is improving as educators become more sophisticated users and programmers. Newer programs are often more challenging to the student user and make better use of the unique capabilities of the computer. We are especially pleased when we receive donations of this type and take special pride in being able to make them available to other teachers through the SOFTSWAP.

Evaluation of Commercial Software

The newest project in the Microcomputer Center is the development of an exemplary collection of commercially produced educational software. We have contacted publishers of microcomputer software with requests for review copies of their educational packages. To date the response has been very encouraging. Many publishers appear to welcome the opportunity to put their materials on display for evaluation. CUE members are previewing and evaluating this software during the summer and the project will continue throughout the coming school year, with tentative plans to publish a collection of reviews. Visitors to the center may preview the software and are invited to help with the review process. However, this software may *not* be copied.

Our goal is to make selected software available for each microcomputer system in the center in order to demonstrate the special features of the system. We also want to give the potential user some idea of the range and quality of materials being published in various subject areas. Finally, we will provide the opportunity for educators to preview a wide variety of software before making purchase decisions. Since it is currently very difficult to find reliable critical reviews of computer software, and because many teachers prefer to make their own evaluation before purchase, we believe that this will be an especially valuable service.

We are also developing a collection of software to demonstrate administrative applications of microcomputers. We will preview and display software designed for school administrators, including word processors and data base management programs. We are also especially interested in potential uses of this type of software in special education programs.

As part of our commitment to encouraging the development of good critical evaluations of microcomputer software, the Microcomputer Center is a participating member of MicroSIFT, a Northwest Regional Educational Laboratory project funded by the National Institute of Education. The objective is to produce and disseminate a high-quality evaluation instrument and to publish reviews of computer software evaluated in terms of the criteria developed. CUE members, under the coordination of the Microcomputer Center staff, have already completed the first round of MicroSIFT program reviews.

Summary

Any new project is accompanied by often unexpected and urgent needs. More than the usual number of critical needs seem to surface with an educational microcomputer project – perhaps because there is apt to be a lack of readily available support resources. The partnership of CUE and the San Mateo County Office of Education has created the Microcomputer Center to meet many of these needs. We have provided educators with a place to see and try various microcomputer systems where other educators who understand their concerns are available. We have created a demonstration site for commercial software, where teachers can "try before they buy," and where they can develop the ability to do critical and objective evaluation of software. We are providing free, educational software and a way for teachers to exchange their own programs with others. We hope we have created a model that will work in other areas and thus will play an even more significant role in developing uses of microcomputers in education. ■

lum development programs and in-service training are the responsibility of LeRoy Finkel, computer consultant, and of Walter Smithey, math-science consultant, working under the direction of Tom Quinn, Director of Curriculum. Dr. James Fee, Director of Media Services, offers assistance with hardware applications and demonstrations. An extensive program to interface computers and microcomputers throughout the county's schools is being developed by Dr. Al Grossman, Administrator of the Office of Information Services. The entire project is under the administration of Dr. Don Halverson, Assistant Superintendent for Educational Support and Planning. This team approach by the county office staff, coupled with the expertise and commitment of Computer-Using Educators, has been crucial to the successful development of the Microcomputer Center.

Manufacturers of microcomputer hardware have also been generous in their support of the center. Tandy (Radio Shack) and Commodore (PET) provided the first microcomputer systems and thus initiated the support that has made possible all subsequent development. Each firm set up a complete system with microcomputer, dual-disk drives, printer and cassette-recorder, all on long-term loan of the Microcomputer Center. A library of commercial programs loaned to the center by each firm includes educational software and word processing systems.

Other firms providing equipment for the center including APPLE, Atari, Bell & Howell and Compucolor. All of the systems are on long-term loan; the center has not purchased any hardware. CUE members were of great assistance in contacting the various firms and securing their support and cooperation.

All of the manufacturers have been helpful in maintaining the equipment in good condition, despite heavy use in the center. They also provide new models as they are developed. This strong commitment on the part of the manufacturers has been another critical factor in the success of the center.

Development of the SOFTSWAP

The SOFTSWAP began as an exchange of instructional programs at the Asilomar Math Conference in 1979. Vince Contreras, San Jose State University professor, organized this first attempt. The programs contributed by participants were deposited in the SMERC Library in the spring of 1980. The first of the equipment being loaned to the center arrived from the manufacturers just as school was out. Interested educators worked throughout the summer to evaluate, edit and revise the programs from the Asilomar Conference.

CUE members served as an Advisory Committee for the Microcomputer Center and also supervised the processing of SOFTSWAP programs for each computer. The committee consisted of Jim Love (PET), Brian Sakai (TRS-80), Dave Stone and Bob Enenstein (APPLE), Marge Fitting (Compucolor), and Pat Tubbs (Atari). The chairpersons also conducted informal demonstrations and evaluation sessions during the summer. As we enter our second year of the SOFTSWAP project, Leslie Grimm is the new chairperson for APPLE; the other systems continue under the direction of the original Advisory Committee members.

Public support of the SOFTSWAP is encouraging. A large number of volunteers continue to donate programs, time and expertise to this growing collection. As new programs flow into the center they are routed to the CUE volunteers for evaluation, debugging and processing. Finished programs are placed onto SOFTSWAP dissemination disks which can be copied without charge by anyone who visits the center.

The large number of requests from outside the San Francisco Bay area led to the mail-order policies adopted this spring. The first catalog was published in the March 1981 *CUE Newsletter* and listed over 200 programs on 12 disks. Fc [illegible]onal disks have been completed, [illegible] total number of programs in the SOFTSWAP to nearly 250. Anyone who sends an original program on a diskette as a contribution to the SOFTSWAP can request a free disk of programs in exchange. Disks are also sold for the cost of the disk, handling and postage. During the first three months of operation over 370 disks have been sent out, some going as far away as Buenos Aires, Singapore and Paris.

SOFTSWAP Procedures

What is different about the SOFTSWAP? Isn't it just another users' group? It is a users' group, but the primary distinction is one of focus – SOFTSWAP concentrates on the development of educational software. Most programs are short, stand-alone instructional units. Many are drill-and-practice exercises written for the elementary school or for remedial work at the secondary level. About one-third of the programs are math oriented. Most games have been excluded since these are generally available from other sources. Only a few utilities, those of special interest to teachers, were on the first series of SOFTSWAP disks; these also are readily available from many user groups.

Every program is evaluated by at least two educators before it is added to a dissemination disk and made available for copying. Programs are edited for spelling errors, inaccurate or incomplete instructions, errors in factual content, programming errors and other problems. Our goal is to distribute only those programs which have some educational value and which are as free of errors as possible in both content and programming. The actual quality of the programs in the collection varies greatly, as does program length. Each disk contains from 12 to 30 programs, all for one system, but includes various subjects and grade levels.

Donors complete a *Preliminary Inventory Form* that identifies and describes the program, and a *Release Form* that gives CUE permission to distribute the program. The donor's form also asks whether the program is original, a modification of another program, or an adaptation from a magazine or book listing.*

The contributed program is put onto a receiving disk where it will remain in its original contributed form as a permanent record. Filled receiving disks are placed into our permanent archives after all of the programs on that disk have been processed One donated program at a time is loaded from the receiving disk onto a working disk and all subsequent versions of that program will be stored on the same disk until the program is finished and ready to be transferred onto the dissemination disk. We have approximately 40 working disks, each with only one program, being evaluated at any given time. At least two educators will review the program on the working disk and make needed revisions. When they agree that it is finished the program will be removed from the working disk and added to the current dissemination disk in the SOFTSWAP. Another program from the receiving disk is then loaded onto that working disk and the evaluation process is repeated. All finished programs on the dissemination disks are available for copying by visitors.

Two forms are used in the evaluation of program: *Checklist for Microcomputer Program Revision* and *Programmer Revision Sheet*. These forms list common errors and problem areas which evaluators are to check. Detailed notes describing needed revisions are made on the *Programmer* sheet and the final reviewer checks to see that all requested corrections have been made. The finished program is described on a final *Inventory Form – SOFTSWAP* and then added to the SOFTSWAP catalog. Newly developed programs are also listed periodically in the *CUE Newsletter*.

These evaluation forms are not designed to produce critical reviews for publication but to provide a guide to be followed in polishing the programs for classroom use. Programs are checked for accuracy of content, clear instructions, trapping of input errors, etc. Certain problems which frequently appear in con-

*Samples of the forms used for the SOFTSWAP are available from the authors at the San Mateo County Office of Education, 333 Main St., Redwood City, CA 94063.

__ Not Accepted __Needs Major Revision __Needs Minor Editing __Finished; add to collection

CHECKLIST FOR MICROCOMPUTER PROGRAM REVISION

Name of program:______________________________

Name of cassette or disk (if different from the program:______________________________

Reviewer's Name:______________________________Date:______________

Position: __Student (age___) __Administrator
__Teacher __Other:______________________________

OKAY	NEEDS WORK	Write in comments as necessary. Make specific notacions on yellow Programmer Revision Sheet.
		A. INSTRUCTIONS
		1. Are they given or are you told how to get them?
		2. Can they be bypassed if desired? Or are they short enough to be shown automatically?
		3. Can user return to instructions during program?
		4. Worded for the program's user group (age, class, etc.)?
		5. Complete and clear?
		B. INPUT
		1. Is there an indicator to show where input will appear?
		2. Do you get hints when you make input errors?
		3. Does program respond to illegal inputs?
		a. Are error messages easy to understand?
		4. Can you "fall out" of the program if you give bad input (e.g., just pressing RETURN)?
		5. Does the input system allow for mistakes to be corrected by user?
		C. OVERALL IMPRESSION
		1. Is the format neat (no words running over the end of the line, lines double spaced for younger users, screen not crowded, etc.)?
		2. When comments, instructions, etc. appear on the screen for a limited time, is that time long enough to be read and understood?
		3. Does the program clear the screen at the start?
		4. Does the package make use of motivational devices? a. Timing b. Scoring c. Graphics d. Effective personalization (informal, conversational, addressing user by name, etc.) e. Random reinforcement f. Free of demeaning remarks
		5. Does the program have difficulty levels appropriate for its purpose?
		6. When appropriate, is correct answer given after a reasonable time or a given number of attempts?
		7. Does the program allow starting over easily?

OKAY	NEEDS WORK	
		C. Overall Impression, cont'd 8. Is there a standardized instruction to continue (e.g., hit spacebar) at bottom of the page? 9. Is program accurate? a. Spelling, punctuation, grammar, etc.? b. Is content accurate? 10. Is it easy to end the program? Is it done neatly (screen clean)?

Please circle the letter that most nearly indicates your opinion of this program.

SUMMARY EVALUATION

A. Level of interest:

interesting A B C D F uninteresting

B. Ease of use (consider student,
teacher, and setting up for next user):

easy to use A B C D F awkward

C. Educational content and/or value:

much A B C D F little

D. Use of graphics:

excellent use A B C D F no use

E. Use of computer delivery:

effective, can't be cone as well by any other means A B C D F there are better ways to achieve this objective

OVERALL VALUE:

every school should have this program A B C D F not worth the effort to load it

Identify the strengths and weaknesses of this package. Please provide a paragraph summary of your reactions to the program. Thank you for being part of our evaluation team.

CONDUIT

COMPUTING IDEAS FOR EDUCATION

CONDUIT
100 Lindquist Center
The University of Iowa
P O Box 388
Iowa City, IA 52244
(319) 353-5789

Director James W
Associate Director Harold

WHAT IS CONDUIT?

CONDUIT is a nonprofit organization that reviews, tests, packages, and distributes educational computer software. Begun in 1971 with support from the National Science Foundation (NSF), CONDUIT is currently affiliated with The University of Iowa and funded in part by NSF and the Fund for the Improvement of Postsecondary Education.

Above all, CONDUIT seeks to distribute quality materials. Our authors are typically college or university faculty who have created materials based on their own teaching experiences and who have used those materials with their students. All our packages are peer reviewed by leading educators for conceptual validity and instructional usefulness. They are also technically reviewed for program accuracy and transferability to a wide range of computer systems.

The CONDUIT library currently includes 148 packages in biology, chemistry, economics, education, English, geography, languages, management science, mathematics, music, physics, sociology, political science, psychology, and statistics. These materials offer a variety of approaches to instructional computing. Several packages are computer simulations of situations that are impossible or too expensive to reproduce in a laboratory. Other packages help teach quantitative techniques in such courses as linear algebra, calculus, operations management, and quantum mechanics. Some of our materials provide problem-solving tools or data for student research. We also distribute packages which tutor students in key concepts or provide drill and practice in basic skills.

A typical CONDUIT package consists of a microcomputer program (for Apple, TRS-80, PET, or Atari) or a program written for standard mainframe computers (in Basic or Fortran); a student manual telling of objectives and methods of use; an instructor's guide illustrating course use; and notes describing installation of the materials on the computer. Although our packages are intended primarily for higher education, a number of high schools have also begun using some CONDUIT programs.

In addition to distributing courseware, CONDUIT is continually studying ways to make instructional computing more effective. Our present activities include research and development of authoring aids, programming standards, transfer guidelines, and evaluation tools. In conjunction with this research, we publish a biannual magazine, Pipeline, featuring articles on new uses of computers in education. We also publish two reports on CONDUIT standards: the CONDUIT Author's Guide describing how to design, develop, and package instructional software materials and the CONDUIT Basic Guide describing how to write and transfer programs in various Basic dialects.

CONDUIT
P.O. Box 388, Iowa City, Iowa 52244
319-353-5789

CONDUIT Package Evaluation Form
For
Microcomputer-Based Instructional Materials

Package Title: ______________________________

Reviewer: ______________________________

Address: ______________________________

Area Code + Phone number ______________________________

I. Summary Assessments

1) Do you recommend the use of this package?

______strongly recommend

______recommend

______recommend subject to improvements (stated in section V)

______do not recommend

Briefly explain your recommendation, identifying the strengths and weaknesses of the package.

2) How central is the subject matter of this package in your field?

______critical, absolutely essential

______important to include

______optional, appropriate but not essential

______trivial, not important

Comment:

3) Is it reasonable to use the computer with this package?

______yes

______no

______not sure

Comment:

II. Description:

1) Describe the topics or concepts presented in this package in a few words. (For example: The law of demand in microeconomic theory at the elementary level.)

2) Suggest the title of a course or courses for which this material is appropriate.

3) Check the appropriate instructional level for using this package: (Check more than one, if appropriate)

___ high school
___ undergraduate (lower level)
___ undergraduate (upper level)
___ graduate
___ other (Specify:

4) What prerequisite skills are needed by students before they use the materials?

5) State the instructional objectives of this package.

III. Course Use:

1) Have you used the package with students?

___ Yes ___ No

This section should be completed only if you have used the package with students.

2) For each course in which you used this package, please state:

Course title: ______________________
Level: ______________________
Enrollment: ______________________

Course title: ______________________
Level: ______________________
Enrollment: ______________________

Course title: ______________________
Level: ______________________
Enrollment: ______________________

3) Describe your students' reactions to these materials.

4) Did your students attain the learning objectives you specified above in section II, question 5?

___ Yes ___ No

If not, can you determine why they didn't?

5) State any suggestions you have for succesfully using the package with students.

IV. Evaluation of Programs:

Please rate the programs on each of the following characteristics.

Exceptional	Very good	Neutral	Fair	Poor	Not applicable	A. Substantive Content
						1. Definition of key concepts
						2. Discussion of underlying assumptions
						3. Validity of principles, theories
						4. Discussion of relevant literature
						5. Overall quality of substantive content

B. Documentation/Textual Materials

						1. Clarity of information in textual materials
						2. Completeness of instructor guide
						3. Adequacy of instructions for operating programs
						4. Overall quality of documentation

C. Support of the Teaching Process

						1. Ease of integration with course procedure
						2. Potential for improving instructor's ability to communicate principles and theories
						3. Potential for improving instructor's ability to communicate methods and techniques
						4. Potential for teaching how to interpret
						5. Overall instructional quality

Exceptional	Very good	Neutral	Fair	Poor	Not applicable	D. Stimulation of Student Interest
						1. Potential for capturing student interest
						2. Challenge to student creativity
						3. Student choice in patterns of use
						4. Appropriateness for student-initiated wor
						5. Overall contribution to student motivatio

						E. Use of Graphics
						1. Motivational value
						2. Direct instructional value
						3. Aesthetic quality of graphics

						F. Use of Color
						1. Motivational value
						2. Direct instructional value
						3. Aesthetic quality of color
						4. Avoidance of distractive use of color

						G. Use of Audio
						1. Motivational value
						2. Direct instructional value
						3. Aesthetic quality of audio
						4. Avoidance of distractive use of audio

Exceptional	Very good	Neutral	Fair	Poor	Not applicable	
						H. Use of Animation
						1. Motivational value
						2. Direct instructional value
						3. Aesthetic quality of animation
						4. Avoidance of distractive use of animation
						I. Screen Layout
						1. General ease of reading layouts
						2. Attractiveness of layouts
						3. Avoidance of overcrowding
						4. Clarity of presentation
						5. Provision for student control of screen transitions
						J. Student Interface
						1. Clarity of prompts
						2. Availability of help within program
						3. Handling of errors in student input
						4. Provisions for student to edit own input
						5. Acceptance of abbreviated input where appropriate
						6. Reporting of performance to student

V. Recommendations for Improvements:

For any characteristic you rated "fair" or "poor" in section IV, please recommend improvements to correct the problem.

VI. Summary:

Please provide a written summary of your general assessment of the package. Your summary should elaborate your evaluation of the substantive aspects of the materials.

SECTION 3

Getting Hardnosed about Software: Guidelines for Software Review

Getting the Message Out of the Medium

Once basic documentation and a program for review have been obtained, the important process of evaluation can begin. The computer program is evaluated first, followed by a review of the related documentation and ancillary materials. The evaluation serves only to provide information on which a decision can be reached; it does not use weighted scales or numerical totals to make the decision for you.

The evaluation model presented in this section is both comprehensive and detailed. It should be tailored to meet the specific needs of a given user. In doing such tailoring, however, the user should be careful to ensure that the resulting version of the model considers each of six important steps of the review process.

Step 1. Load the program on your system. Run the program briefly to become familiar with the program's "flow."

Step 2. Execute the program as a successful student would, avoiding intentional or careless errors. Extend the program when possible by interacting "creatively" as a good student would in testing the cleverness of the programmer.

Step 3. Execute the program as an unsuccessful student would. Respond incorrectly to test how the program handles student errors. If an erroneous response to a task results in the repetition of that task, make repeated incorrect responses. Be sure to repeat the *same* response and also try giving *different* incorrect responses. Finally, make other kinds of errors such as typing mistakes, incorrect form of input (e.g., "one" for "1"), content errors, and errors in following the directions.

Step 4. Use the checklist for the evaluation of software on pages 17-18, adapting it as necessary to fit your own needs. Throughout, your responses should be based *only* on your observations and experiences with the program. Once completed, your responses can be compared with the vendor's claims.

Step 5. Complete the checklist for the evaluation of documentation and ancillary materials. Again, the extent of the review will depend on your needs and the impressions remaining from your review of the actual software.

Step 6. Make a decision.

The design of the sample Software Evaluation Checklist calls for items on one side of a sheet of paper and abbreviated explanations of item vocabulary or purpose on the other side of the sheet. (Detailed discussions of the terms used on this sample are presented in the next subsection of these *Guidelines.*) This one-sheet design is judged highly desirable insofar as it simplifies paper shuffling. (Of course, the items and their explanatory notes could be placed on your computer. This not only saves trees and storage space but also illustrates that you recognize another obvious advantage of computers in education.)

Although the sample checklist here purposely includes an excess of items, in practice a shorter form may do the job. Moreover, experienced evaluators would require few, if any, of the explanatory notes given on the back of the sheet.

Words to Evaluate By

The use of our checklist, or any other checklist, by more than one person requires an agreement regarding the meaning of terms used to classify or evaluate programs. We must extend the minilexicon we started in section 2. We'll do this by proceeding item by item through the sample checklist, stopping to define and discuss concepts as the need arises.

1. Instructional Range

For any given program, the appropriate grade level and ability level must be judged according to the particular instructional setting. (A program that is informational at first grade might be remedial at third.) Key factors in determining grade or ability level are the concepts in the program. Other important factors are reading level, prerequisite skills, degree of student control, and intended instructional use. It is possible for a program to be flexible enough to be used across a wide range of grade levels and ability levels.

2. Instructional Grouping

The interaction between the learners and a program can be either direct or through an intermediary (the teacher). The first two groupings described below involve direct interaction; the third uses an intermediary.

Individual. Some programs are designed specifically for use by a single person. Programs of this nature may include a summary of individual performance; others may allow a specific level of difficulty—a specific set of spelling words, for example. Programs for individuals may have provisions for easier or more difficult situations, depending on the individual student's response.

Small group. Some programs are specifically designed for use with two, three, or four students. One such example might be an arithmetic fact game involving a car race in which each student can move forward on a correct answer. Other programs for small groups might be business simulations in which each small group is a business.

Large group. Large-group use of programs may vary in intent. A program may be used with a large group solely for the purpose of introducing the program for later use by individuals or small groups. Some programs are designed for use by a teacher in graphically demonstrating a concept. (Trigonometric functions provide one example; fraction concepts such as equivalence are another.)

3. Execution Time

The time required for the use of a program will vary considerably. The load time of the program depends on the complexity of the program and on whether it is being loaded from a cassette tape or a disk. (The disk is a much faster but more expensive storage medium.) The type of program is a factor (e.g., tutorial versus informational). The user will require additional time if responses are consistently incorrect. Another time factor is the degree of user control (i.e., how many options are available to the user). An estimate given as a range is an appropriate response for this item.

4. Program Uses

The classification of programs by the instructional uses to which they are put is difficult. Careful analysis notwithstanding, definitions seem doomed to fuzziness, and a given program may seem to fit in more than one category. Moreover, the use of a given program can often be determined by the classroom practitioner; a program written with one use in mind may be used in completely different ways by each of several persons. In fact, a given person may identify more than one use for a program, leading to the necessity to indicate *primary* and *secondary* uses.

Despite the difficulties just noted, there is instructional value in attempting to identify the use or uses of a program. The following definitions are proffered as an aid to the classification of those program uses that appear on the checklist. Additional uses are described later in this section.

Drill or practice: Programs that assume previous instruction in the concept, skill, or process to be addressed. They present a controlled sequence of exercises designed to drill the recall of certain pairings (e.g., to associate French verbs and their English counterparts or to recognize musical notes aurally) or to practice some algorithm (e.g., to add two four-digit whole numbers, to sort buttons on the basis of two or more criteria, or to assemble a telephone).

Tutorial: Programs through which the computer assumes total responsibility for instruction. Tutorial programs are characterized by a dialogue between the student and the computer in which the direction and level of the dialogue are shaped by student input. Ideally, such programs would be high fidelity simulations of the best teaching behavior associated with a given topic and would control the variables associated with the what, why, when, and who of the instructional episode.

Simulation: Programs that attempt to represent key aspects of some environment within which the user will experience the necessity to make decisions and will be informed of the results of those decisions *without experiencing the real consequences of possible misjudgments.* The time required to develop and use simulations with high fidelity is justified in situations where actual experience is ruled out because of extreme expense, safety considerations, or the time required for the actual experience. Simulations include problem-solving tasks (e.g., negotiation of a bank loan, diagnosis of illnesses or or equipment failures, genetic experiments, testing theoretical models), procedural tasks (e.g., acid titration, blasting, the breeding of organisms), and performances (e.g., control of water pollution).

Instructional gaming: Programs calling on the user to apply one or more specific skills or concepts in a game environment. The game dimension of the environment includes the conditions of competition with oneself or others, specific (if arbitrary) rules, the need to develop a winning strategy, and the introduction of random events to force the revision of strategies. This means that the primary objective of the exercise should be the development of general problem-solving skills. This objective is often accompanied by the objective of exercising specific concepts or skills in a new context. Examples of games that have been computerized include chess, Master Mind, and nim.

Problem solving: Programs serving either the student or the instructor; the problem to be investigated may be either within a given subject or within instruction in this subject. In practice, there are two levels of use—one that employs existing programs using known algorithms (e.g., the distance traveled by an object in free-fall during a given period of time) and one that requires the development of a program employing a user-defined algorithm (e.g., a program that finds the sum of distances from proposed locations of six warehouses to each of the establishments to be supplied by those warehouses). In fact, the development of the algorithm itself may be the object of the problem-solving activity.

Informational: Programs designed to generate information (lists of prime numbers, decimal approximations of rational numbers correct to any given number of decimal places, powers or roots of a given number, synonyms for any word from a given list, a list of formulas or rules, etc.). Although such programs might be employed to do problem solving or to construct a program in some other mode, the face value of the program remains that of data generation. They offer very little interaction with the person for whom the data is generated.

A blank at the end of item 4 of the checklist invites the user to specify other program uses. Another six possible program uses are suggested below.

Demonstration: Programs that assume the intervention of an informed manager (teacher) between the program and the learner. They are designed to permit the manager to pursue the questions and suggestions of students by varying the conditions associated with some concept being developed. (For example, it might permit the variation of weather conditions in investigating their effects on annual production of selected crops.) These programs are generally devoid of textual display relative to their purpose and operation. This lack of text, while precluding direct use of the program by the learner, permits efficient realization of the program's purpose by an informed user (teacher). External documentation is critical to demonstration programs.

Instructional management: Programs designed to aid the user in the flexible and efficient management of such limited instructional resources as time, equipment, workspace, instructional personnel, and consumable materials. Such programs assume, at least in part, the tasks of making instructional diagnoses, giving course assignments, evaluating student progress, keeping student records, and facilitating communication among instructors and students.

Instructional support: Programs designed to facilitate the improvement of instructional products (e.g., worksheets, laboratory activities, newsletters, reports to students or parents), resource location (e.g., listings of publishers, suppliers, materials, or human resources), and product selection (e.g., textbook readability, software evaluation).

Test construction and analysis: Programs to aid in criterion-referenced testing or tailored testing. These programs offer such services as the banking of objectives or test items, item analysis, record keeping, test generation, and test scoring.

Programming utility: Program building blocks that may be used to construct instructional programs. Their availability allows instructors or programmers to make efficient use of their hardware and their programming time. Examples of such building blocks would include programs that search a student's input for spelling, alphabetize lists, search lists for synonyms, create or delete data files, recall and display selected geometric figures, and translate typed input into spoken words.

Whistles and bells: Programs designed to acquaint the user with those characteristics of a computer system that may have instructional application. Such programs range from those that carefully demonstrate and explain a feature like screen protection (allowing sustained display of a chart or diagram without the possibility of student erasure or write over) to those that mindlessly exercise color and sound for the purpose of selling hardware.

5. User Orientation: Instructor's Point of View

Flexibility. A program may permit the user or instructor to adjust it to accommodate a range of ability levels, several degrees of difficulty with respect to a given user, or the class of concepts involved. This can be accomplished through options presented by the program. For instance, a program dealing with the practice of whole number computation might offer a choice of the operation (addition, subtraction, multiplication, division) or the range of numbers involved (0–10, 0–20, 0–100). Another option might involve the difficulty of the problems presented in terms of the procedure required for a solution (i.e., borrowing versus no borrowing in the subtraction problems). Programs that do not offer such options to the student may have a provision for the execution of such adjustments by the teacher.

Intervention or assistance. Although it is a desirable goal for most programs to operate independently of the classroom teacher, there are programs for which assistance is needed and even desirable. For instance, some drill programs are "unending." Intervention to stop the program is done by the teacher and involves a judgment of what constitutes sufficient program use by a given student. (*Note:* A rating of "low" on "freedom from need to intervene or assist" would mean that considerable teacher intervention or assistance is required.)

6. User Orientation: Student's Point of View

Quality of directions. Directions need to be concise and clear. They should be developed for the lowest level of use. Some directions can be simplified by the use of an example.

Quality of response. Responses to student input should be low-key and understandable. Responses to incorrect input should be neutral (e.g., "Incorrect. Try again." versus "Wrong, dummy! How could you get this far and know so little?").

Screen formatting. Formatting refers to the physical layout of text and graphics presented on the screen. Examples of poor formatting include a full screen of text with single spacing, too many graphics, continually flashing displays, and text scrolling off the screen. Some good examples are centering of data, a few items on the screen with the user in control of when the next items appear, and reverse-field printing of key items (i.e., light print against a dark screen versus dark print against a light screen).

Need for external information. A program may require the user to have access to information other than that printed by it. Generally speaking, the more self-contained a program is, the better. However, certain programs intentionally require the use of external items. Examples of such programs include programs dealing with dictionary skills and the use of maps.

Disruption by system error. System errors result in the involuntary termination of a program. They may be introduced by improper calculations, errors in the logic of the program, or input that is conceptually correct but does not meet the form required by the program. Programs written to prevent all systems errors—syntactic errors and improper input—are the most desirable. They can be developed by using special input routines.

Simplicity of student input. A program should ensure that a user knows when and in what form input is needed. Characters with special meanings should be avoided. Input locations on the screen should be standardized throughout the program. Typing requirements should be minimal. If a special type of input is required, an example should be given.

7. Content

Instructional focus and significance. The program's topic should be clearly defined. The instructional objectives of the program must be viewed as important by the instructor. Also, the program should represent a valid use of the computer's capabilities. (The program should not repeat the same set of questions or present problems in exactly one sequence.)

Validity and compatibility. The concepts and terms used should be correct, clear, and precise. The rate of presentation should be consistent with the levels for which the program is intended. The content, terminology, teaching style, and educational philosophy should be consistent with those generally encountered by the student.

8. Motivation and Instructional Style

There are many options that may increase student motivation or develop a particular instructional style. Some of these options are a function of the hardware being used. Because these features are optional, a column labeled "none," meaning the feature is not present, has been added at the left of the low-high continuum.

Student involvement. The proper degree of involvement will vary with the type of program used. A drill and practice program for one student is highly active, whereas an information program may require little student input. Tutorial programs, being dialogues, should require student involvement.

Student control. Several concepts may be involved here. Options available to the student at the beginning of the program may include different kinds of content, levels of difficulty, or the type of response the student may wish to use. Additional examples of student control include the selection of the number of problems to be done and the provision of help (at the student's request) at any point in the program.

Game format. The use of a game format often generates favorable responses from students. Some examples are mathematics drill as a drag race or in a tic-tac-toe or concentration format. Caution must be exercised so that the game format does not overshadow the instruction. Note that a game format does not imply that the program use is instructional gaming; a drill program may use a game format. A focus on problem-solving strategies is required for a program to be classified as having an instructional gaming use.

Graphics, animation, color. The functional use of one or more of these features can enhance instruction; their improper or irrelevant application can interfere or cause misconceptions. The uses of graphics, animation, or color must be examined carefully to ensure that they support concept development. One might begin that examination by asking what would be lost if the feature were deleted.

Voice and nonvoice audio. The use of voice input and output has considerable potential. Be aware that the quality and use of voice is dependent on the hardware. The use of nonvoice audio provides an additional means for attracting interest but may be distracting in the classroom.

Light pen. Some programs require the use of a light pen. This can be a good feature if a light pen is available, but a program employing a light pen will be useless without it.

Ancillary materials. Materials external to the program may be used. Some may be provided, but their use may not be required (e.g., pretests or posttests, worksheets, etc.). Others may be necessary but are locally available (e.g., dictionaries or local road maps). Care should be taken to note any materials developed especially for the program that must be supplied by the program's author or publisher.

As in item 4, a blank is left at the end of item 8. This is necessary because the list of special features grows daily; its length is a function both of the subject matter and the computer system involved. The listed features are in relatively common use; other examples include touch panels (the user can execute an instructional task simply by touching the screen with a finger), special keyboards for the physically handicapped, electronic devices that monitor a physical phenomenon (such as water temperature) and use it in the program's execution, and mechanical devices (robots, if you will) that interact with the program.

9. Social Characteristics

Competition and cooperation. Some programs may be written for use by a single student, several students, or the whole class. Competition may be between two or more students or between a student and the computer; it is generally motivating. The need for cooperation among students, also desirable, might be found in a business simulation that requires several students to operate the business.

Humanizing the computer. Programs may give the computer human characteristics. Some people find this desirable, whereas others wish to emphasize that the computer is a machine. Asking for the student's name may or may not be necessary or desirable. Older students generally find it boring, distracting, or phony.

Moral issues. Items or techniques that at first create interest may be questionable from other points of view. Capital punishment as exhibited by most hangman games is one example; war games are another.

Summary of student performance. A performance summary is a feature inherent to games that result in a winner. Performance on a simulation (e.g., the use of a credit card) also calls for a summary evaluation (e.g., the credit card was cancelled because of the inability to meet payments regularly). Drill programs often conclude with a summary regarding the speed or accuracy of the student's performance. Depending on the manner in which these summaries are presented, they may have either a positive or a negative effect on the learner.

Notes on the Design or Redesign of the Checklist

With the exception of the few descriptive requests at the top of the form and in items 1 through 4, the use of the checklist requires only making checks, circling words, or bracketing ranges of dots on judgment scales. The design of the scales in items 5 through 9 was with an eye to consistency; it also consistently expresses the summary prejudices of the writers. For each scale, a rating toward the left-most extreme (identified by such terms as "low," "passive," and "poor") would generally be, for the writers, a negative evaluation of the software. (This technique isn't foolproof. A rating of "low" on "freedom from need for external information" might be desirable for a drill program but is probably neither desirable nor possible with a simulation.) That is, the writers felt that a program should rank *high* in flexibility, should call for *active* student involvement, should make *good* use (if any) of color, and should employ competition in a *positive* way only. This left-is-negative and right-is-positive arrangement of the scales allows one to get the general sense of an evaluation by scanning the completed checklist. When items are added or altered on the checklist, it would be a good idea to keep this design intact. Thus, should you feel that program control should be the purview of the teacher or that judgments should be within the program itself, you might reword the second part of item 8 to read as follows:

low *high*
* * * * * freedom from student control

as rewritten, the word *low* would continue to be associated with an undesirable characteristic.

SOFTWARE EVALUATION CHECKLIST

PROGRAM NAME: ______________________ SOURCE: ______________________ COST: __________

SUBJECT AREA: ______________________ REVIEWER'S NAME: ______________________ DATE: __________

1. INSTRUCTIONAL RANGE

______________________ grade level(s)

______________________ ability level(s)

2. INSTRUCTIONAL GROUPING FOR PROGRAM USE

____ individual

____ small group (size: ____)

____ large group (size: ____)

3. EXECUTION TIME

__________ minutes (estimated) for average use

4. PROGRAM USE(S)

____ drill or practice

____ tutorial

____ simulation

____ instructional gaming

____ problem solving

____ informational

____ other (______________________)

5. USER ORIENTATION: INSTRUCTOR'S POINT OF VIEW

low				high	
•	•	•	•	•	flexibility
•	•	•	•	•	freedom from need to intervene or assist

6. USER ORIENTATION: STUDENT'S POINT OF VIEW

low				high	
•	•	•	•	•	quality of directions (clarity)
•	•	•	•	•	quality of output (content and tone)
•	•	•	•	•	quality of screen formatting
•	•	•	•	•	freedom from need for external information
•	•	•	•	•	freedom from disruption by system errors
•	•	•	•	•	simplicity of user input

7. CONTENT

low				high	
•	•	•	•	•	instructional focus
•	•	•	•	•	instructional significance
•	•	•	•	•	soundness or validity
•	•	•	•	•	compatibility with other materials used

8. MOTIVATION AND INSTRUCTIONAL STYLE

passive				active	
•	•	•	•	•	type of student involvement

low				high	
•	•	•	•	•	degree of student control

none	poor				good	
•	•	•	•	•	•	use of game format
•	•	•	•	•	•	use of still graphics
•	•	•	•	•	•	use of animation
•	•	•	•	•	•	use of color
•	•	•	•	•	•	use of voice input and output
•	•	•	•	•	•	use of nonvoice audio
•	•	•	•	•	•	use of light pen
•	•	•	•	•	•	use of ancillary materials
•	•	•	•	•	•	use of ______________

9. SOCIAL CHARACTERISTICS

present and negative	not present	present and positive	
______	______	______	competition
______	______	______	cooperation
______	______	______	humanizing of computer
______	______	______	moral issues or value judgments
______	______	______	summary of student performance

1. The grade levels and ability levels for a particular program are primarily determined by the concepts involved. Other important factors are reading level, prerequisite skills, degree of student control, and intended instructional use. It is possible for a program to be flexible enough to be used across a wide range of grade levels and ability levels.

2. Some programs are designed for use by individuals. Others have been or can be modified for participation by two or three persons at a time. Simulations or demonstrations often pose opportunities for large-group interaction. A given program may be used in more than one grouping, depending on the instructor.

3. The time required for the use of a program will vary considerably. Include loading time for cassettes. A time range is the appropriate response here.

4. Instructional programs can be categorized according to their uses. Some programs may have more than one use, thus falling into more than one of the following categories:

Drill or practice: Assumes that the concept or skill has been taught previously.

Tutorial: Directs the full cycle of the instructional process; a dialogue between the student and the computer.

Simulation: Models selected, alterable aspects of an environment.

Instructional gaming: Involves random events and the pursuit of a winning strategy.

Problem solving: Uses general algorithms common to one or more problems.

Informational: Generates information (data).

5. These are factors relevant to the actual use of the program from the point of view of an instructor.

Flexibility: A program may allow the user or the instructor to adjust the program to different ability levels, degrees of difficulty, or concepts.

Intervention or assistance: A rating of "low" means considerable teacher intervention or assistance is required.

6. These are factors relevant to the actual use of the program from the point of view of a student.

Directions: The directions should be complete, readable, under the user's control (e.g., should not scroll off the screen until understood), and use appropriate examples.

Output: Program responses should be readable, understandable, and complete. If in response to student input, the output should be of an acceptable tone and consistent with the input request.

Screen formatting: The formats during a program run should not be distracting or cluttered. Labels and symbols should be meaningful within the given context.

External information: A program may require the user to have access to information other than that provided within it. This may include prerequisite content knowledge or knowledge of conventions used by the program designer as well as maps, books, models, and so on.

System errors: System errors result in the involuntary termination of the program.

Input: A program should ensure that a user knows when and in what form input is needed. It should avoid using characters with special meanings, restrict input locations to particular screen areas, and require minimal typing.

7. These are matters relevant to the subject-matter content of the program.

Focus: The program topic should be clearly defined and of a scope that permits thorough treatment.

Significance: The instructional objectives of the program must be viewed as important by the instructor. Also, the program should represent a valid use of the computer's capabilities while improving the instructional process.

Soundness or validity: The concepts and terms employed should be correct, clear, and precise. Other important factors are the rate of presentation, degree of difficulty, and internal consistency.

Compatibility: The content, terminology, teaching style, and educational philosophy of the program should be consistent with those generally encountered by the student.

9. Competition, cooperation, and values are concerns that may be a function of the way a program expresses them. (War gaming and the "hangman" format are sample issues.) Also, the "humanizing" of the computer may serve for motivation or to reduce anxiety, but it also may become tedious, misleading, and counterproductive.

The summary of student performance can be dichotomous (win or lose), statistical (time expended or percent of items correct), or subjective (as in the evaluation of a simulation). It may be for student, teacher, or both.

SCHOLASTIC SOFTWARE EVALUATION FORM

The following "Scholastic Software Evaluation Form" is an exerpt from MICROCOMPUTERS IN EDUCATION: A SCHOLASTIC IN-SERVICE TRAINING PROGRAM by James Poirot and Karen Billings, published by Scholastic Book Services in 1982. The package contains a set of materials designed for educators to conduct their own computing workshops: a notebook for the planner/presenter, a set of participant handbooks, and slides for use by the presenter. The notebook contains objectives, lecture notes, workshop management suggestions, forms, and activities for the following topics: Computers in Education; Software Evaluation; Hardware Evaluation; Programming in BASIC; Computer Literacy; Future of Technology in Education; and Planning for the Future. A section on resources lists books; periodicals; films; software publishers, software catalogs, and sources of software reviews; hardware manufacturers; and organizations and associations. In addition, purchase of the package includes ten 1-year subscriptions to ELECTRONIC LEARNING magazine. For more information, write Scholastic, Inc., 730 Broadway, New York, New York 10003.

SCHOLASTIC SOFTWARE EVALUATION FORM *

I. IDENTIFICATION

A. Program Name: ____

Single Program ____ Series ____

B. Distributor:

Name: ____

Address: ____

____

Telephone: ____

C. Microcomputer

Type: ____

Memory Required: ____ 16K ____ 32K ____ 48K ____ 64K

Special Language Required: ______________________________

Storagu Medium: ____ Tape Cassette ____ 5'' Diskette ____ Module

____ Other: ____________________

Equipment Requirements: ____ One Disk Drive ____ Two Disk Drives

____ Color ____ Printer ____ Voice/Sound Instrument

____ Game Paddles ____ Other: ____________________

II. INSTRUCTION

A. Grade Level

K 1 2 3 4 5 6 7 8 9 10 11 12

B. Software Types

____ Drill and Practice
____ Tutorial
____ Simulation

____ Tool
____ Diagnostic/Prescriptive
____ Management
____ Other: ____________________

C. Curriculum Areas

____ Language Arts
____ Social Studies
____ The Arts
____ Business Education
Mathematics
____ Reading

____ Science
____ Career Education
____ Foreign Languages
____ Physical Education
____ Other: ____________________

D. Instructional Considerations

____ Game Approach
Student Workbook Required

____ Classroom-Text Dependent
____ Teacher Supervision Required

*From MICROCOMPUTERS IN EDUCATION: A SCHOLASTIC IN-SERVICE TRAINING PROGRAM, by James Poirot and Karen Billings, Scholastic Book Services, 1982.

E. Special Characteristics

____ Timing

____ Branching

____ Sound Capabilities

____ Student Records

____ Material Modification

F. Software Instruction Time

____ Average Per Lesson

____ Average Per Package

III. EVALUATION

For each criteria listed below, circle the appropriate Weight and Rating value. Multiply each Weight by each Rating to obtain a Product. Then add together the Weights and add together the Products to obtain the Total for each.

A. Presentation Criteria

	Not Important	Somewhat Important	Important	Very Important	Extremely Important	Poor	Fair	Good	Very Good	Excellent	
	Weight					Rating					Product
Ease of Use	1	2	3	4	5	1	2	3	4	5	
Reliability	1	2	3	4	5	1	2	3	4	5	
Motivation	1	2	3	4	5	1	2	3	4	5	
Frame Display	1	2	3	4	5	1	2	3	4	5	
Documentation	1	2	3	4	5	1	2	3	4	5	
	1	2	3	4	5	1	2	3	4	5	

Total ____ Total ____

B. Content Criteria

	Weight					Rating					Product
Accuracy	1	2	3	4	5	1	2	3	4	5	
Feedback	1	2	3	4	5	1	2	3	4	5	
Level of Difficulty	1	2	3	4	5	1	2	3	4	5	
Appropriateness for Computer Use	1	2	3	4	5	1	2	3	4	5	
Educational Standards	1	2	3	4	5	1	2	3	4	5	
	1	2	3	4	5	1	2	3	4	5	

Total ____ Total ____

To obtain an Average for Presentation Criteria, divide the Total of the Products by the total of the Weights. Repeat the process for Content Criteria. Add together the Average of Presentation Criteria and the Average of Content Criteria. Finally, divide the sum of the two Averages by **2** to obtain the Final Rating.

Presentation Criteria: Total of Products/Total of Weights = Average
____________ / ____________ ______

Content Criteria: Total of Products/Total of Weights = Average
____________ / ____________ ______

Final Rating: (____________ + ____________)/2 = ______

IN MANY WAYS, EVALUATING microcomputer software is like evaluating any other kind of instructional material. You'll be looking for materials that are: • content-accurate; • appropriate for the grade level; • interesting; • free of bias; • and accompanied by thorough and well-organized teachers' guides.

But in evaluating microcomputer software, there are other aspects to consider that you won't encounter in other instructional materials. For example, since computers are interactive, you'll want to see if the software makes use of the computer's uniquely interactive capabilities in ways that both hold students' attention and also instruct them.

You'll also want to keep in mind that a program which excites inexperienced student-users today, may bore a more computer-literate class next year. "CAI has gotten a lot of praise as a medium because kids find it interesting," says Mark Falstein, editor of *Courseware Report Card*, a new software review journal. "But as computers become part of our lives, it'll be old hat." In other words, Falstein says, programming should not rely on the novelty of computer delivery to captivate students, but rather on the interest level of the material and the manner in which it is presented.

To help you get started in evaluating microcomputer software, we've provided a form on pages 48 and 49. The form represents a synthesis of the many evaluation forms available today (see note below), and of comments and concerns expressed most frequently by educators and instructional computer experts. The form can be helpful in two ways. It can help you evaluate a piece of software that you are previewing for possible purchase. And, once filled out, it can serve as a "hard-copy" record of your review, to be included in a school or district software library.

Most of the form is self-explanatory. But, for the first-time user, here are some things to keep in mind as you use the form.

The Software Line-up

What Reviewers Look For When Evaluating Software

USING THE EVALUATION FORM

PART I OF THE FORM ASKS FOR SOME GENERAL BACKGROUND INFORMATION about the program being reviewed. Be sure to mention any special hardware the program requires, such as a voice synthesizei or joysticks, and any special software, such as a language not standard on the microcomputer intended for use.

In Question 2, list any skills the program assumes knowledge of, e.g., does a math program require students to know how to graph?

For Question 6, briefly discuss the appropriateness of the computer as a vehicle for teaching the program's curriculum content and objectives. For example, you might

(Continued)

Software Lineup

(Continued)

note: "Inappropriate; computer serves only as an electronic workbook," or "highly appropriate use of the computer for simulation."

Part II of the form includes an Evaluation Checklist. Here, again, there are a number of questions to pay close attention to. For example, Question 3 under the Educational Content section—"Is the difficulty level consistent for material, interest, and vocabulary?"—is the downfall of many otherwise commendable programs. "I saw a program whose thrust was reading problems in math," says J. R. Pennington, principal, Decatur High School, Decatur, GA. "The program was beautifully done. But while the math was fifth-grade, the language was about first year of college!"

A PROGRAM THAT EXCITES INEXPERIENCED STUDENT-USERS TODAY, MAY BORE A COMPUTER LITERATE CLASS NEXT YEAR.

The Presentation section assesses the manner in which the program conveys the content. In answering Question 4—"Do graphics, sound, and color, if used, enhance the presentation?"—consider whether "flash" is substituted for well-thought-out content.

In the Interaction section, note question 1—"Is feedback effective and appropriate?" Be wary of such feedback responses as "Wrong, dummy." "That's a totally unacceptable response," says Mark Falstein. Question 4 asks about student control of the pacing and sequencing of information. Good programs will allow students to control the pace at which information is presented.

In the Teacher Use section, pay special attention to the question about documentation (Question 4). Good documentation will cover most problems that a teacher may encounter in using a program. It will also be written in "plain English."

One final note. All review forms, ours included, will affect the way you look at a program. "The instrument tailors the response," says Stan Silverman, a New York teacher. "I once gave teachers different evaluation instruments and had them look at the same programs. I got all different responses from them. I try to teach teachers to look at the programs, and not be misled by forms."

Most computer-using educators agree, however, that an evaluation form, like the one on the next two pages, does have its place. It can provide you with some general guidelines for reviewing new programs; it can help you build your own set of criteria for judging software; and it can provide you with a handy record of programs in your software library.

Editor's Note: The following educational organizations, institutions, and magazines have developed forms for use in the review and evaluation of microcomputer software: (1) California Library Media Consortium for Classroom Evaluation of Microcomputer Courseware, San Mateo County Office of Education, Redwood City, CA; (2) *The Computing Teacher*, University of Oregon, Eugene, OR; (3) *Curriculum Review*, Palo Alto, CA; (4) EPIE Institute and Vicki L. Blum, Stony Brook, NY; (5) Microcomputer Resource Center, Teachers College, Columbia University, NY; (6) MicroSIFT, Northwest Regional Educational Laboratory, Portland, OR; (7) The National Council of Teachers of Mathematics, Reston, VA.

A District Software Evaluation Plan that Works

SOFTWARE EVALUATION IS one problem that elementary school educators in Utah's Granite school district don't have to wrestle with. The district office in Salt Lake City takes care of it for them.

Bob Ives, a soft-spoken man with seven yea[illegible]xperience in instructional computing, heads the software evaluation program at Granite. The district is Utah's largest, including as it does the state's only large metropolitan area. There are 60 elementary schools in the district, staffed by 1,200 teachers, and attended by more than 35,000 students. More than 200 micros are at their disposal, and, according to Ives, "That number is growing every week."

One of Ives' tasks is keeping track of the vast number of instructional programs made available each year. He says he hears about such programs in various ways—primarily through ads in magazines, and sometimes through other teachers.

If a program sounds promising, Ives tries to get a copy to preview. "We've had good luck getting preview copies," he says. "I get them either from the companies, or from neighborhood computer stores."

Most of these programs are then evaluated by a member of a reviewing committee, which consists of two teachers from different grade levels and two school principals. Whenever possible, the program is reviewed by a teacher from the program's intended grade level. But other programs are reviewed by people outside the committee who just want to help out.

A standard three-part form is provided for the evaluation. Part One asks for basic information; Part Two asks more specific questions about the program's content and organization; and Part Three asks the teacher for a more subjective discussion of the program's strengths and weaknesses. "We devised parts One and Three here," Ives says; "Part Two I took from a journal for gifted students."

Completed reviews are filed in a review library which consists of some 200 programs. About four times a year, Ives meets with computer education facilitators—there is one at each elementary school—to keep them abreast of what's available and what's worthwhile.

The computer facilitators in the Granite district serve as a liaison between the district office and individual schools. Teachers at the elementary schools who are looking for software first turn to their facilitators; then, when they narrow down the programs they're interested in, they request an evaluation from the district office. The decision on whether or not to purchase the software is then made by the staff at the schools, rather than by the district office.

Granite's software evaluation program is already a model of simplicity and effectiveness, but Ives still isn't satisfied. This year, he's planning to put software evaluations on diskettes to be made available quarterly to schools with micros. Soon, he says, software evaluations may also be available on a data base being assembled at the University of Utah —on-line for schools with modems, and in hard-copy form for others.

—*R. Neumann*

Software Evaluation Form

Reviewer's Name:____________________ Date of Review:____________

Address/Phone:____________________( ) ____________

Program Title____________________ Medium:___5" disk;___8" disk;

___cartridge; ___tape

Package Title____________________ Copyright Date (if any)____________

Microcomputer (brand, model, memory)____________________

Necessary Hardware____________________ Necessary Software____________

Producer____________________ Author(s)____________

Back-up Copy Policy ____________________Cost____________

PART 1

Program Overview and Description

1. Subject area and specific topic____________________
2. Prerequisite skills necessary____________________
3. Appropriate grade level (circle) 1 2 3 4 5 6 7 8 9 10 11 12 college
4. Type of program (check one or more)

___Simulation
___Educational Game
___Drill and Practice
___Tutorial
___Problem Solving
___Authoring System

___Testing
___Classroom Management
___Other (specify)____________________
____
___Remediation
___Enrichment

5. Appropriate group instructional size: ___individual___small group ___class
6. Is this program an appropriate instructional use of the computer?____________________

____

7. Briefly list the program's objectives. Are they clearly stated in the program or in the documentation? Are they educationally valuable? Are they achieved?

____

____

____

8. Briefly describe the program. Mention any special strengths or weaknesses.

____

____

____

PART 2

Evaluation Checklist

Please check Yes, No, or Not Applicable for each question below. To add information, or to clarify an answer, use "Comments" at the end of each section.

Yes	No	N/A	EDUCATIONAL CONTENT
____	____	____	1. Is the program content accurate?
____	____	____	2. Is the program content appropriate for intended users?
____	____	____	3. Is the difficulty level consistent for material, interest, and vocabulary?
____	____	____	4. Is the program content free of racial, sexual, or political bias?

Comments: ____

____

Yes	No	N/A	PRESENTATION
____	____	____	1. Is the program free of technical problems?
____	____	____	2. Are the instructions clear?
____	____	____	3. Is the curriculum material logically presented and well organized?
____	____	____	4. Do graphics, sound, and color, if used, enhance the instructional presentation?
____	____	____	5. Is the frame display clear and easy to read?

Comments: ____

____

Yes	No	N/A	INTERACTION
____	____	____	1. Is the feedback effective and appropriate?
____	____	____	2. Do cues and prompts help students to answer questions correctly?
____	____	____	3. Can students access the program "menu" for help or to change activities?
____	____	____	4. Can students control the pace and sequence of the program?
____	____	____	5. Are there safeguards against students "bombing" the program by erroneous inputs?

Comments: ____

____

Yes	No	N/A	TEACHER USE
____	____	____	1. Is record-keeping possible (within the program or through documentation worksheets)?
____	____	____	2. Does teacher have to monitor student use?
____	____	____	3. Can teacher modify the program?
____	____	____	4. Is the documentation clear and comprehensive?

Comments: ____

____

PART 3

Overall Evaluation

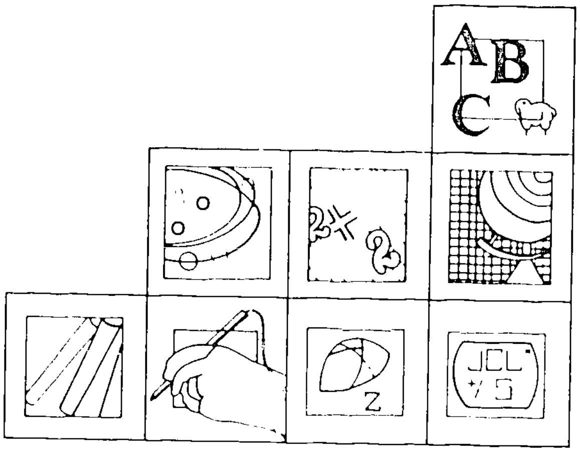
CHECK ONE.

___ Excellent program. Recommend without hesitation.

___ Pretty good program. Consider purchase.

___ Fair. But might want to wait for something better.

___ Not useful. Do not recommend purchase.



Software Reviews

Three samples of software reviews are presented on the following pages. The reader will note a number of contrasts among the reviews: the number of categories of evaluation, the amount of description, the formats for presenting evaluation summaries, and the relative emphasis on selected features of software differ considerably among these samples. Because the information in software reviews can differ from source to source, consumers of software evaluation data might do well to examine several different published reviews, in light of the information needed to inform local software decisions. In addition, it may be possible to find several different reviews for the same software.

Each of the sample reviews is the product of an organized system for making evaluations of software available to educators on a regular basis. These systems share the following characteristics:

- program description
- category schemes that logically group program features
- criteria for evaluation of software features
- rating procedures and/or forms to summarize evaluative information
- specific procedures for conducting evaluations
- processes to train and maintain groups of professional software evaluators
- systems to maintain files of and/or publish and distribute evaluations of selected educational software

The first example, MicroSIFT, represents a nonprofit institutional approach involving an extensive network of organizations and individuals concerned with software evaluation. *School Microware Reviews* represents a commercial organization that operates an evaluation system dependent on evaluative input from participating educators. *Courseware Report Card*, another commercial system, employs staff and consultants to develop software evaluations.

All of the systems represented make their evaluations available to educators. MicroSIFT makes their software evaluations available through the network of cooperating institutions, and through a public access database called Resources in Computer Education (RICE). *School Microware Reviews* and *Courseware Report Card* publish collections of reviews at regular intervals, which are available on a subscription basis. *School Microware Reviews* also makes its evaluations available to software publishers and encourages dialogue between reviewers and publishers.

Each of the sample evaluations on the following pages provides a glimpse of a specific software product. As such, they are a good starting point in the process of deciding what software to use in the classroom. But no single review can provide all the information necessary to decide if a particular product is the right one for its intended use. These evaluations can help educators identify potential selections that can be examined further at the local level by software review committees and individual teachers. Permission to reproduce the information in this section was granted by the publishers.

Arithmetic Racing

PRODUCER: Math Software
1233 Blackthorne Place
Deerfield, IL 60013

LOCAL DISTRIBUTORS: Contact producer for list

EVALUATION COMPLETED: Fall 1981; Revised 2/1/82

VERSION: © 1980

COST: Not sold individually; sold in packages of 5 to 10 programs ranging from $100 to $250

ABILITY LEVEL: Grades 4-11
SUBJECT: Mathematics: speed and accuracy drill of arithmetic operations
MEDIUM OF TRANSFER: 5" flexible disk
REQUIRED HARDWARE: 32K Apple II or II Plus, one disk drive, monitor
REQUIRED SOFTWARE: DOS 3.2 or 3.3, Applesoft
INSTRUCTIONAL PURPOSE: Standard instruction, enrichment
INSTRUCTIONAL TECHNIQUES: Game, problem solving, drill and practice
DOCUMENTATION AVAILABLE: Suggested grade level, program operating instructions, demonstration

INSTRUCTIONAL OBJECTIVES: To improve students' arithmetic skills in addition, multiplication, subtraction, and division; to provide an interesting interactive environment for remediation work in arithmetic facts; and to develop speed and accuracy in working basic arithmetic operations.

INSTRUCTIONAL PREREQUISITES: The program assumes that students know basic arithmetic facts concerning the operations of addition, multiplication, subtraction and division. Students also need to understand the rules governing the operation of the computer game.

CONTENT AND STRUCTURE: ARITHMETIC RACING is a game of timed arithmetic practice for students Grades 4-11. Players first select addition, subtraction, multiplication, or division and then specify the largest number they want the computer to give them. Players also select a speed level from 1-5. The computer then assigns a point value to each problem based on these selections. A 25 point bonus is added to the score for answering each of the ten questions correctly.

ESTIMATED STUDENT TIME REQUIRED: Not available

POTENTIAL USES: The program may be used in a classroom setting to provide drill and practice in basic arithmetic operations.

MAJOR STRENGTHS: The game format offers an incentive for students needing drill and practice in arithmetic operations.

MAJOR WEAKNESSES: None cited.

EVALUATION SUMMARY

SA	A	D	SD	NA	
	●				Content is accurate.
●					Content has educational value.
				●	Content is free of stereotypes.
	●				Purpose of package is well defined.
	●				Package achieves defined purpose.
	●				Content presentation is clear and logical.
	●				Difficulty level is appropriate to audience.
●					Graphics/sound/color are used appropriately.
	●				Use of package is motivational.
	●				Student creativity is effectively stimulated.
	●				Feedback is effectively employed.

SA	A	D	SD	NA	
	●				Learner controls rate and sequence.
	●				Instruction integrates with prior learning.
	●				Learning can be generalized.
	●				User support materials are comprehensive.
	●				User support materials are effective.
	●				Information displays are effective.
	●				Users can operate easily and independently.
	●				Teachers can employ package easily.
	●				Computer capabilities are used appropriately.
	●				Program is reliable in normal use.

SA - Strongly Agree A-Agree D-Disagree SD - Strongly Disagree NA - Not Applicable

Evaluators indicate they would use or recommend use of package with little or no change.

Northwest Regional Educational Laboratory
300 S.W. Sixth Avenue • Portland, Oregon 97204
(503) 248-6800

This evaluation is based on the evaluations of three or more reviewers who are representative of potential users of the courseware package.

Permission to reproduce this document is hereby granted.

SCHOOL MICROWARE EVALUATION FORM (COPY AS NEEDED)

Your Name__________ Organization__________ Position__________

Address__________ Tel:__________

Product Name What's Different Supplier PDI Price $16.95 No. of Progs. Under This Name 10 segments

Subjects and Grades to Which Applicable Language Arts-Reading Comprehension Est. Student Time Required__________

FUNCTIONAL DESCRIPTION - Describe the program in terms of its goals and what it does to achieve them. Give as much detail as possible (use extra sheets as necessary).

The student is given four words, three of a similar group of things and one which does not belong in the group. He types in the number of the word that is different from the others. Each segment requires making increasingly difficult choices. Vocabulary words are grade 2 or lower (Cont. Over)

PRELIMINARY CONSIDERATION - Does this program contribute to the teaching of topic(s) which should be taught in today's schools? ___Yes ___No If No, give your reasons for this answer in the Comments section at the end of the form and omit the balance of the questionnaire.

INSTRUCTIONS - Enter a number in the blank to indicate the extent to which the product fulfills the description in the item, as follows: 2 - Completely, 1 - Partially, 0 - Not at All. If the item is not applicable to the product, enter N/A. If the item is unclear, enter U. Elaborate on answers as necessary in Comments at end or on extra sheets, giving item numbers.

DOCUMENTATION - List materials accompanying the program, e.g., teachers guide, student workbook.

1. Indicate types of information included.
   - 2 a. Suggested course/subject, grade levels.
   - 2 b. Goals.
   - 2 c. Performance objectives.
   - 1 d. Suggested teaching strateg(ies).
   - 0 e. Correlation with standard texts.
   - 1 f. Prerequisites for use of program.
   - 0 g. Student exercises, teacher answers.
   - 1 h. Operating instructions.
   - 0 i. Listing and sample runs of program(s).
   - N/A j. If a simulation, description of the model used.
   - 0 k. Suggested topics for follow-up discussions.
   - 0 l. Suggested references/activities for follow-up.

2 2. The documentation is written clearly.

N/A 3. If a workbook is included, the format and content are appropriate.

INSTRUCTIONS GIVEN TO USER BY PROGRAM

1. The instructions are adequate regarding:
   - 2 a. The instructional task to be performed.
   - 1 b. Details of how to interact with the program.

0 2. User has the option of skipping instructions if already known.

STUDENT-COMPUTER DIALOG

2 1. Output is displayed screen by screen (paged) rather than scrolled.

2. If output is paged:
   - 2 a. User has control over continuing to the next page.
   - 2 b. Amount of information in each page is appropriate.
   - 2 c. The perceptual impact (amount of type and lines) is suitable.

2 3. Output is spaced and formatted so as to be easily readable.

2 4. Language is well suited to most students' reading ability.

2 5. Uses correct grammar, spelling, hyphenation and punctuation.

N/A 6. Any grid or coordinate system used is consistent with common conventions.

2 7. Students can respond with common symbols & ways of using them, e.g., right to left entry of sums.

2 8. Accepts abbreviations for common responses.

2 9. Provides for individual needs, e.g., opportunity to work with harder or easier material.

0 10. Dialog is personalized, i.e., makes appropriate use of student names.

0 11. Uses devices to get & maintain interest, e.g., variation of computer responses, humor, pace change, surprise.

12. Makes good use of any special features computer:
   1 a. Graphics N/A b. Color N/A c. Sound

2 13. Reinforcing responses (indications of right, wrong, etc.) are appropriate.

2 14. The number of wrong answers allowed is reasonable.

N/A 15. Responds appropriately if allowed number of wrong answers is exceeded.

N/A 16. Provides opportunity to get help if difficulty is encountered.

2 17. Minimizes bad entries via devices such as objective formats (multiple choice, etc).

N/A 18. Deals well with inappropriate entries, i.e., response to typing errors, etc., is intelligible and useful.

2 19. Required entries are within students' capabilities (esp. typing, vocabulary).

2 20. Reports student performance periodically and at end of session.

MISCELLANEOUS CONCERNS

N/A 1. If a simulation, the program gives a sufficiently accurate representation of the situation simulated.

2 2. The concepts and vocabulary required to use the program are reasonable.

2 3. Operates properly and is free of bugs.

2 4. Is well structured and documented internally to facilitate any necessary debugging/modification.

COMMENTS - Please use this space and additional sheets as necessary to provide any information which you believe would help someone who is thinking about buying of the product being reviewed. In particular, indicate what you like best and least about the program. Also, list any changes which should be made.

The stated goal of this program is to build reading comprehension and logic skills. It would be more accurate to say that the program improves vocabulary and logic skills. Students from 4th to 8th grade have en-

Revised 8/82 (Cont. Over)

Functional Description (Continued)

in game one and gradually work up to grade 6 or lower in games 8-10.

If a student gives a wrong answer the program requires him to try again until he gets the problem right. Every wrong answer is counted in the score.

One very nice feature in the program is that after the student gives the right answer the program tells him why his answer is right (e.g. 1. red, 2. blue, 3. clear, 4. green -- 3 is correct because clear is not a color).

At the beginning of the segment the student has an opportunity to choose the number of the question with which he wishes to start, allowing him to bypass questions on the lower segments. The user can also stop before the end of the segment and receive his score up to that point.

Comments (Continued)

joyed using this program and it provides good practice in classifying words. The only drawback is one of format. Every question is first marked off inside a box and there is a slight delay before the words appear in the box. Some students find the waiting somewhat irritating. A little more interaction in the way of personalizing the program and rewarding the user would be welcome. The student should be told that his score will appear at the end of the segment and that every wrong answer will count, not just one per problem.

COURSEWARE report card 150 West Carob Street • Compton, CA 90220

Subject Area: Social Studies

Grade Level: 8 and up

Type of Program: Simulation

System Requirements:

- TRS-80, 48K, disk
- Apple II, 48K, disk (DOS 3.3)
- ATARI, 40K, disk; 32K, tape

Price: $24.95

Publisher:

Creative Computing Software
39 E. Hanover Avenue
Morris Plains, NJ 07950

Hail to the Chief

Creative Computing Software

Simulation of an election campaign in which the student becomes a candidate for president. The student must declare positions on real-life campaign issues and use program-generated opinion polls to determine the strategy necessary to win the election.

It's a rough business, being a presidential candidate. At a news conference you're asked to clarify your position on energy policy. Some sharp reporter points out that the position is inconsistent with the one you took last week. Attempting to recoup, you lay out $150,000 from your carefully budgeted campaign fund for a nationally televised address on military preparedness. The public's response? "An actor could have done better!"

These are two of the pitfalls to be avoided in *Hail to the Chief*, an outstanding simulation of a presidential election campaign from Creative Computing Software. The student-player becomes a presidential candidate running against a program-generated opponent. The program incorporates such real-life political concerns as public image, regional interests, campaign spending limits, the advantage of incumbency, issues, and debates. The "candidate" is able to see the results of decisions reflected in national opinion polls and to plan strategy accordingly

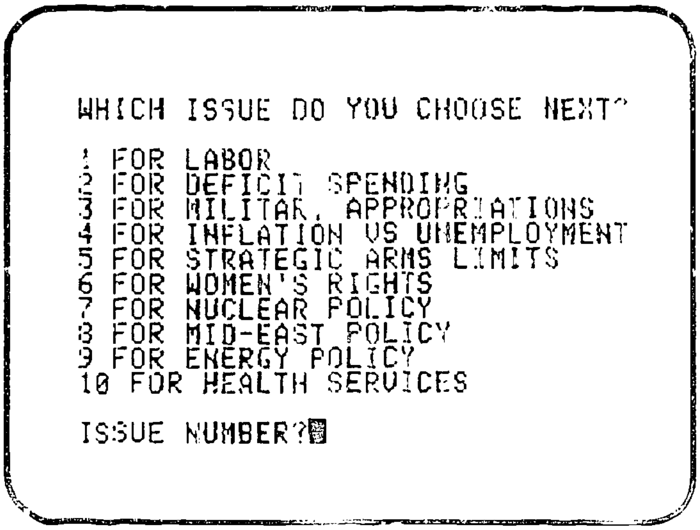
Description of the Program

The game can be played at any of four levels of complexity. Levels 2 and up assign the player to either the Democratic or Republican party; at level 1 there is no party assignment. Levels 3 and 4 add the factor of incumbency; either the player or the programmed opponent is the sitting president or is supported by the incumbent. The higher levels also add the problems of campaign funding and staying within spending limits. At each level, the player may choose from among ten degrees of difficulty. The higher the difficulty number, the harder it is to win: the player may be pitted against a popular incumbent or assigned by the program to the Democratic party in an era of Republican conservatism.

The player must take positions on ten campaign issues *(see Figure 1)*, each defined by a brief statement in the program manual. The player takes a position on each issue by entering numbers ranging from -70 to +70. (It is perhaps indicative of the viewpoint of the program's creators that the negative numbers indicate liberal positions while positive numbers indicate conservative positions. An entry of zero represents a middle-of-the-road position.) At level 1 the player enters positions on all ten issues at the start of the game. At higher levels the player initially declares positions on only three issues but will be asked to declare positions on other issues during the game.

Performance:	B+
Ease of Use:	A
Error Handling:	A
Appropriateness:	A
Documentation:	C
Educational Value:	A

The program simulates a nine-week campaign, from September 1 to Election Day, Tuesday, November 4. The program proceeds by increments of several days; each new date is displayed on the screen. In each time-frame, as they used to say in the Nixon White House, the player must select a particular campaign strategy to pursue *(see Figure 2)*. These include televised appearances, regional ad cam-

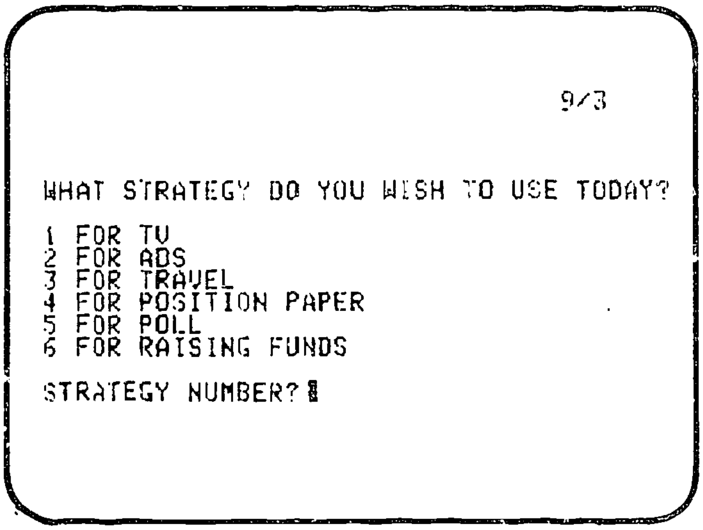


```
WHICH ISSUE DO YOU CHOOSE NEXT?

1 FOR LABOR
2 FOR DEFICIT SPENDING
3 FOR MILITAR, APPROPRIATIONS
4 FOR INFLATION VS UNEMPLOYMENT
5 FOR STRATEGIC ARMS LIMITS
6 FOR WOMEN'S RIGHTS
7 FOR NUCLEAR POLICY
8 FOR MID-EAST POLICY
9 FOR ENERGY POLICY
10 FOR HEALTH SERVICES

ISSUE NUMBER?█
```

Figure 1. *Player must take positions on ten campaign issues. The issues are defined by statements included in the program manual.*



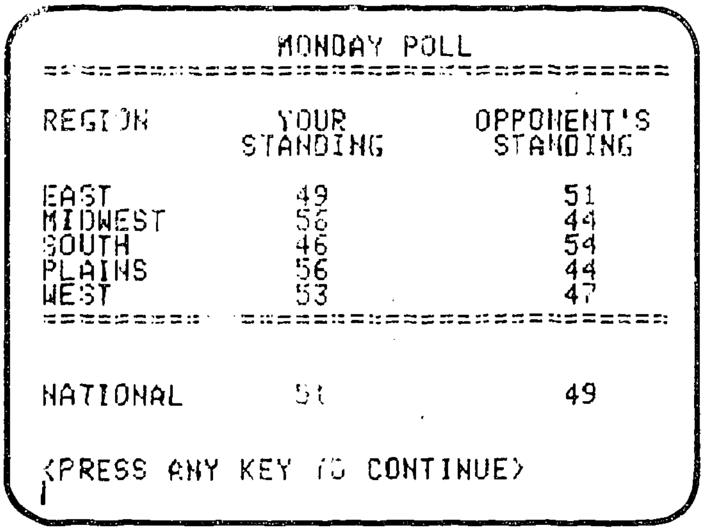
```
                                         9/3


WHAT STRATEGY DO YOU WISH TO USE TODAY?

1 FOR TV
2 FOR ADS
3 FOR TRAVEL
4 FOR POSITION PAPER
5 FOR POLL
6 FOR RAISING FUNDS

STRATEGY NUMBER?█
```

Figure 2. *The six campaign strategies available to the player. "Current date" is displayed at upper right.*



```
                MONDAY POLL
=======================================

REGION        YOUR          OPPONENT'S
            STANDING         STANDING

EAST           49               51
MIDWEST        56               44
SOUTH          46               54
PLAINS         56               44
WEST           53               47
==========  ===========================


NATIONAL       51               49


<PRESS ANY KEY TO CONTINUE>
```

Figure 3. *The Monday poll.*

paigns, speaking tours, position papers, taking a national poll by issue and region, and (on level 4 only) fund raising. In addition, the program will randomly throw in such events as debates and news conferences in which the player must answer questions on the issues. In some cases the program gives immediate feedback on the results of strategy: "A very poor showing"; "A quite effective appearance." More specific results are shown in the straw poll *(see Figure 3)* displayed every "Monday," showing the candidates' standings in five regions of the country: EAST, MIDWEST, SOUTH, PLAINS, and WEST. (The manual includes a tabulation of electoral vote by state and region so that the player may plan regional strategies.) The effective learning experience of the game is in the player's analyses of these results in order to plan subsequent strategy for the campaign.

The following example illustrates how this works: A candidate identified as a relatively liberal Democrat makes a national TV appearance. The program asks him to select two issues on which to state his position (on the -70/+70 continuum). The computer judges it "A quite effective appearance." The following Monday poll, however, shows an erosion of support in the conservative SOUTH and PLAINS regions. Now the candidate chooses to take an in-depth poll that shows region-by-region opinion on each issue. Through this poll the candidate determines that his positions on military spending and energy are more consistent with opinion in the SOUTH and PLAINS than his positions on other issues. He then may select making a speaking tour of those regions, or, if funds are short, taking out ads in those regions, clarifying his positions on those particular issues. Sure enough, the next Monday poll shows his standing in the SOUTH and PLAINS to be markedly improved.

In the next time increment, the program declares a news conference. Our candidate is asked his opinion on Saudi oil. He responds with a "+20," reflecting his relatively conservative energy policy. However, this response is seen as being inconsistent with his strongly pro-Israel Middle East stance. "WAFFLER!" the program declares, "YOUR POSITIONS ARE INCONSISTENT!" The next poll shows an erosion of support in the generally pro-Israel EAST. The candidate may then choose to travel to the EAST to reaffirm his support of labor and of government-supported health care.

The program continues in this manner until Election Day, when final results are computed and displayed and electoral vote broken down by region. A losing candidate is subjected to a funeral dirge, while a winner is treated to the strains of "Hail to the Chief" and the display "CONGRATULATIONS, PRESIDENT-ELECT."

83

Evaluation

Performance: The program runs smoothly. The student has as much time as necessary to read and analyze each display. Material remains on the screen until the student types the called-for response and/or presses RETURN. The program takes forty to sixty minutes or longer to run, however, and the length of time it takes for the computer to analyze responses will test the patience of some students. A nine-week campaign may parallel the real thing, but in this format, especially at the lower levels of complexity, it tends to be repetitive.

Ease of Use: Limited only by a student's reading ability or understanding of the concepts. All required responses are in the form of one- and two-digit numbers. Menus and instructions make the meaning of each response clear. If a student is confused, typing HELP will call up an explanation.

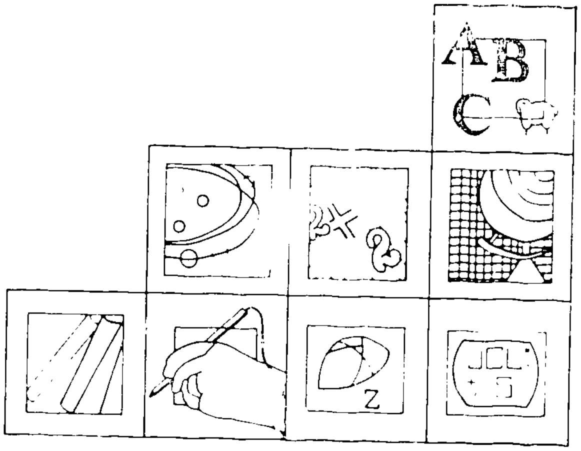
Error Handling: We reviewed the Apple version and found it impossible to crash the program short of hitting the RESET key. Any inappropriate response is met simply with a reminder to USE NUMBERS and with a repetition of the question.

Appropriateness: In our view, this is what computer-based instruction should be about. No textbook or filmstrip can possibly deal with American electoral politics in such a dynamic and interactive way. The program's creators have included an impressive array of situations and variables that could be incorporated into such material only by means of a computer. Sound is limited to a few brief musical phrases and a beep or two; it should not be terribly distracting in a classroom. Apple and ATARI versions include a color display of the White House with the title; the remainder of the program is entirely text. While this makes for humdrum video, the older students, for whom the program is intended, should be able to take it. Memory-eating color graphics might have cut into the program's content.

Documentation: The eight-page manual explains the basic rules of the game quite adequately, but some of the game's important features have to be learned through trial and error. For a program that takes so long to run, we wish that the instructions could warn a player to be sure to write down the numbers entered for positions on the issues, and those given for the opponent. *Any* departure from these initial positions is regarded as waffling. Worksheets with space to record this information might have been included, as well as a few hints on campaign strategy as it affects the flow of the game. As it stands, a player can waste an hour or more of class time figuring out how the game works.

Educational Value: The program's incorporation of many real-life situations and its balance between direct feedback to student responses and the presentation of data requiring interpretation make *Hail to the Chief* the best simulation program we've ever seen. There are no arbitrary gains or losses, as in such highly promoted programs as *Oregon Trail.* The player is able to see, directly and indirectly, the consequences of his or her decisions. The various levels of complexity allow the game to be played by students over a range of grade levels and abilities. Instructions both on-line and in the manual are clear and free of errors. This program is an effective and dynamic supplement to any unit on American government and the electoral system. The only reservation we have is that it is *too* real a simulation: The public relations and media manipulation required to win are almost as Machiavellian as in an actual election campaign. Principles and ethics do not enter into this simulation at all; image and cleverness are the only relevant values. The program's response to a player's bad TV showing is "An actor could have done better!" Students will recognize that in real life, one did.

Resource Information



Books

Burke. R.L. *CAI Sourcebook*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1982.

Includes a section on "CAI Courseware Review."

Coburn, et al. *Practical Guide to Computers in Education*. Reading, MA: Addison-Wesley, 1982.

Contains a discussion of educational software selection.

Criteria for Evaluating and Selecting Microcomputer Courseware. Baltimore, Maryland: State Department of Education, 1982.

Douglas, Shirley and Gary Neights. *A Guide to Instructional Microcomputer Software* and *A Guide to Microcomputers*. Pennsylvania Department of Education, reprinted by Connecticut Department of Education, 1980.

Introduction to selecting software and implementing microcomputer-based instructional systems.

Edwards, Judith B., et al. *Computer Applications in Instruction: A Teacher's Guide to Selection and Use*. Hanover. NH: Time Share Corporation, 1978.

Introduction to computers in education, including instructional software selection.

Northwest Regional Educational Laboratory, *Evaluator's Guide for Microcomputer-based Instructional Packages*. International Council for Computer Education. Eugene, Oregon: 1982.

The guide uses MicroSIFT's evaluation form, describes the process, and gives detailed instructions for using the form.

Intentional Educations, Inc. *Computers in Education: A Practical Guide*. Reading, MA: Addison-Wesley, 1982.

Includes evaluation criteria and strategies for implementing computer-based instructional systems.

Isaacson, D. *How to Design Educational Microcomputer Programs*. Fresno, CA: California State University, 1981.

A teacher's guide to designing software. Illustrations, sample programs, criteria, etc.

Naiman, Adeline. *Microcomputers in Education: An Introduction.* Chelmsford, MA: Northeast Regional Exchange, 1982.

A beginner's guide to the use of microcomputers in schools. Includes a section on software selection, and sample software evaluation forms.

National Council of Teachers of Mathematics (NCTM). *Guidelines for Evaluating Computerized Instructional Materials.* Reston, VA: NCTM, 1981.

Guidelines for evaluation and a form are included.

Poirot, James. *Computers and Education.* Manchaca, TX: 1980.

Includes a discussion on evaluation of software.

Poirot, James, Kathleen Swigger, and Merridee Heidt. *Evaluation Guide for TABS Related Courseware.* Houston, TX: 1981.

This book is designed to aid Texas teachers in evaluating software related to state-wide assessments of basic skills achievement. It includes an evaluation form which allows reviewers to obtain a quantitative measure of software effectiveness.

Pitts, Marcella. *The Educator's Unauthorized Microcomputer Survival Manual.* Washington, D.C.: Council for Educational Development and Research, 1982.

A beginner's guide which contains questions for reviewing software.

Strohmenger, Todd. *Guidelines for Selecting and Developing Secondary Remediation Software.* Charleston, WV: Appalachia Educational Laboratory, 1983.

In disk format, this publication is designed to aid in assessing software for students with developmental problems other than physical or mental disabilities.

Texas Education Agency. *Guide for Selecting A Computer-Based Instructional System.* Austin, TX: Texas Education Agency, 1982.

Contains guidelines for software selection.

Vann, Eric G. *Microcomputers in the Classroom: A Practical Guide for Educators.* Glen Ellen, IL: Institute for Educational Research, 198

Willis, Jerry and William Danley, Jr. *Nailing Jelly to a Tree: A Guide to Educational Software.* Beaverton, OR: Dilithium Press, 1981.

Directories

The Addison-Wesley Book of Apple Computer Software, 1982
The Book Company
16720 Hawthorne Blvd.
Lawndale, CA 90260

Describes and evaluates all types of Apple software.

American Peripherals
122 Bangor Street
Lindenhurst, NY 11757

Two editions; one lists 1200 educational programs for PET; the other lists 400 for VIC-20.

The Apple Software Directory, Volume Three: Education
WIDL Video
5245 West Diversey Avenue
Chicago, IL 60639

Describes and indexes by subject, Apple educational software from more than 400 vendors.

Atari Program Exchange
Atari,Inc.
P.O. Box 427
155 Moffett Park Drive, B-1
Sunnyvale, CA 94086

Quarterly editions. Atari's compilation of user-written software. Includes special education section.

CIE Software News
Computer Information Exchange
Box 159
San Luis Rey, CA 92068

A newsletter with a continuously updated directory of software, books, and hardware news.

Classroom Computer News Directory of Educational Computing Resources
Intentional Educations, Inc.
341 Mt. Auburn Street
Watertown, MA 02172

Yearly, A reference arranged by category, state, region, type of computer. Lists sources of software reviews.

Commodore Software Encyclopedia
Commodore Business Machines
Computer System Division
Systems Marketing Group
681 Moore Road
King of Prussia, PA 19406

Lists Commodore software in seven categories, including education.

Curriculum Product Review
530 University Avenue
Palo Alto, CA 94301

Lists texts, AV materials, hardware and software.

Educator's Handbook and Software Directory
Vital Information, Inc.
7899 Mastin Drive
Overland Park, KS 66204

Lists evaluated educational programs for the Apple. Includes articles on microcomputer applications in education.

Huntington Computing Catalog
P.O. Box 1297
Corcoran, CA 93212

Lists educational and noneducational programs.

IDEAS
ECS (MR1-1/M40)
Digital Equipment Corporation
200 Forest Street
Marlboro, MA 01752

Lists educational software for DEC mainframe computers.

Index to Computer-Based Learning, 1981 Edition
Anastasia Wang, ed.
Instructional Media Laboratory
University of Wisconsin
P.O. Box 413
Milwaukee, WI 53201

Lists almost 5000 educational programs.

Instant Software
80 Pine Street
Peterborough, NH 03458

For TRS-80, Apple, PET, TI-99/4 and Atari 800 microcomputers.

Instructor's 1982-83 Computer Directory for Schools
Attn: Elsa Silander
P.O. Box 6099
Duluth, MN 55806

Includes articles on software selection and lists of educational software grouped by curriculum area, machine compatibility, and publisher.

International Microcomputer Software Directory
Imprint Software
420 South Howes Street
Fort Collins, CO 80521

Lists microcomputer software in all areas, including education.

K-12 Micro Media
172 Broadway
Woodcliff Lake, NJ 07675

Marck
280 Linden Avenue
Brandon, CT 06405

Lists tested educational programs for Apple, Atari, PET, and TRS-80.

Micro Co-Op Newsletter
P.O. Box 432
West Chicago, IL 60815

Bimonthly newsletter which provides software listings and descriptive comparisons of programs.

Microcomputers Corporation Catalog
34 Maple Avenue
P.O. Box 8
Armonk, NY 1050

Listing of computer accessories and software; many programs are educational.

Minnesota Educational Computing Consortium (MECC).Instructional Computing Catalog
2520 Broadway Drive
St. Paul, MN 55113-5199

A catalog of educational courseware for Apple II and Atari computers.

Opportunities for Learning, Inc.
Dept. L-4
8950 Lurline Avenue
Chatsworth, CA 91311

Elementary through college level.

Queue
5 Chapel Hill Drive
Fairfield, CT 06432

Catalogs educational software for Apple, Atari, PET, and TRS-80.

Radio Shack TRS-80 Educational Software Sourcebook
From: Radio Shack Stores

Describes software available for TRS-80 microcomputers in eleven subject areas.

Reference Manual for the Instructional Use of Microcomputers
JEM Research, Discovery Park
University of Victoria
P.O. Box 1700, Victoria, BC
CANADA V8W 2Y2

Indexes and evaluates educational programs for the Apple II.

School Microware Directory
Dresden Associates
P.O. Box 246
Dresden, ME 04342

Lists and describes educational programs for Apple II, Atari, PET, and TRS-80.

Scholastic Microcomputer Instructional Materials
904 Sylvan Avenue
Englewood Cliffs, NJ 07632

All programs have passed educator evaluation tests.

The Software Directory
Software Central
P.O. Box 30424
Lincoln, NE 68503

Lists and briefly describes programs for various microcomputers.

Sources for Courses
TALMIS
115 North Oak Park Avenue
Oak Park, IL 60301

Annually lists educational programs for Kindergarten through College.

Starbek Software Directory
11990 Dorsett Road
St. Louis, MO 63043

Describes over 1000 Apple-compatible programs.

1982 Swift's Directory of Educational Software, Apple II Edition
Sterling Swift Publishing Co.
1600 Fortview Road
Austin, TX 78704

Describes commercial and noncommercial educational programs for the Apple.

Texas Instrument's Home Computer Program Library, 1982
From: Texas Instruments Dealers

Lists software for the Texas Instruments microcomputer.

User's
2520 Broadway Drive
St. Paul, MN 55113

Lists Apple and Atari Software developed by educators for the Minnesota Educational Computing Consortium (MECC).

VanLoves Apple II/III Software Directory, Vol. II
Vital Information, Inc.
7899 Mastin Drive
Overland Park, KS 66204

A comprehensive directory of Apple software which includes an educational software section.

Articles

Anderson, Ronald, "Courseware Deserves Evaluation Including Peer Review," *AEDS Monitor, 20* (April-June 1982):10-12.

Barry, Tim, "Considering Software Reviews," *InfoWorld, 3* (Nov. 2, 1981):20-22.

Barry, Tim, "Considering Software Reviews II," *InfoWorld, 3* (Nov. 9, 1981):20-21.

Bingham, Margaret, "Software Evaluation: A Coordinated Effort Needed," *AEDS Monitor, 20* (April-June 1982):10-12.

Caldwell, Robert M., "Guidelines for Developing Basic Skills: Instructional Materials for Use with Microcomputer Technology," *Educational Technology, 20* (October 1980):7-12.

Cohen, Vicki Blum, "Criteria for the Evaluation of Microcomputer Software," *Educational Technology* (January 1983): 9-14.

Dyer, Susan R. and Richard C. Forcier, "How to Pick Computer Software," *Instructional Innovator, 27* (September 1982):38-40.

"ERIC Reports," *Educational Technology, 22* (April 1982):61-62.

Forman, Denyse, "Courseware Evaluation for Curriculum Integration," *AEDS Monitor, 20* (April-June 1982).

Green, Doug, "Green's Eleven Commandments for Educational Software," *InfoWorld, 3* (November 23, 1981):5

Grimes, Lynn, "Computers are for Kids: Designing Software Programs to Avoid Problems of Learning," *Teaching Exceptional Children,* (November, 1981):49-53.

Hakansson, Joyce, "How to Evaluate Educational Courseware," *The Journal of Courseware Review, 1* (1982).

Hannaford, Alonzo E. and Florence M. Taber, "Microcomputer Software for the Handicapped: Development and Evaluation," *Exceptional Children, 49* (October 1982):137-144.

Hilgenfeld, Robert, "'Checking Out' Software," *The Computing Teacher* (November 1981).

Holznagel, Donald, "Courseware and Software Needs in Education," *The Computing Teacher, 8* (May 1981):36-38.

Holznagel, Donald, "What Courseware is Right for You?"*Microcomputing Magazine* (October 1981).

Hunter, Beverly, "How to be a Program Critic," *Electronic Learning, 1* (January-February 1982):65.

Judd, Dorothy, and Robert Judd, "Evaluation of Instructional Programs for Microcomputers," *Educational Computer,* 2 (March-April 1982):16-17.

Kansky, Bob, William Heck, and Jerry Johnson, "Getting Hard-Nosed about Software: Guidelines for Evaluating Computerized Instructional Materials," *The Mathematics Teacher, 74* (November 1981).

Koetke, Walter, "Software Shopping," *Kilobaud Microcomputing* (July 1981).

Lathrop, Ann, "Building the Software Collection," *Educational Computer, 1* (November-December 1981):23+.

Lathrop, Ann, "Microcomputer Software for Instructional Use — Where a [illegible] ritical Reviews?' *The Computing Teacher, 1* (September-October 1981):14-15.

Lathrop, Ann, "Software . . . Previewing and Reviewing," *Educational Computer, 1* (Sep October 1981).

"Level-Headed Guide to Software Evaluation," *Classroom Computer News, 1* (July-Augus 22-23.

Moursund, David, "Some Thoughts on Reviewing Software," *The Computing Teacher, 7* (June 1980).

Needle, David, and Deborah Wise, "Helping Teachers Pick Software," *InfoWorld, 3* (Dec. 7, 1981):25.

Peters, Harold J., and Molly H. Hepler, "Reflections on Ten Years of Experience," *AEDS Monitor, 20* (April-June 1982):10-12.

"Quality Software: How to Know When You've Found It!" *Electronic Learning, 1* (November-December 1981):33-36.

Roblyer, M.D., "Guidelines for Evaluating Computerized Instructional Materials," *Educational Technology, 22* (February 1982):44-45.

Roblyer, M.D., "When is it 'Good Courseware?': Problems in Developing Standards for Microcomputer Courseware," *Educational Technology, 21* (October 1981):47-54

Wade, T.E., Jr., "Evaluating Computer Instructional Programs and Other Teaching Units," *Educational Technology, 20* (November, 1980).

Wager, Walter, "Issues in the Evaluation of Instructional Computing Programs," *Educational Computer, 1* (September-October 1981):20

Wager, Walter, "The Software Evaluation Dilemma," *AEDS Monitor, 20* (April-June 1982):5-6.

Watt, Molly, "Making a Case for Software Evaluation," *Computing Teacher, 9* (May 1982): 20-22.

Wise, Deborah, "Software is the Key to Educational Computing, Teachers Say," *InfoWorld, 3* (November 23, 1981:17+.

Clearinghouses and Information Centers

Apple Computer Clearinghouse for the Handicapped
Prentke Romich Company
R.D. 2, P.O. Box 191
Shreve ,OH 44676

A source of information about Apple software being developed for handicapped people.

Apple for the Teacher
c/o Ted Perry
5848 Riddio Street
Citrus Heights, CA 95610

Operates the National Computer-Assisted Library for the Apple, a Software collection of international scope. Publishes a newsletter containing reviews.

Boston Computer Society (BCS)
Educational Resource Exchange
Three Center Plaza
Boston, MA 02108

Maintains an educational resource exchange which disseminates information on software selection.

California Library Media Consortium
San Mateo County Office of Education
333 Main Street
Redwood City, CA 94065

Organized in 1981 by 54 library media specialists. Publishes COURSEWARE REVIEWS. All reviews are written by educators and based on classroom use of software.

California Software Clearinghouse
Office of Staff Development
State Department of Education
721 Capitol Mall
Room 634
Sacramento, CA 95814

The Clearinghouse supports fifteen Teacher Education and Computer Centers (TECC) around the state, provides staff development, print resources, a collection of software and disseminates evalutions of software.

CONDUIT
P.O. Box 388
Iowa City, IA 52244

Tests, reviews and distributes software; the focus is on software for higher education.

Computer Technology Task Force
Superintendent of Public Instruction
#7510 Armstrong St., S.W. #FG 11
Tumwater, WA 98504

This task force operates a telephone exchange for Washington educators which can be reached at 206/753-2858 or 206/753-6747. The task force has also published six handbooks on educational technology. One handbook is devoted to software evaluation.

Computer-Using Educators (CUE)
1776 Educational Park Drive
San Jose, CA 95133

Operates a software library, a microcomputer demonstration center, and an in-service training program.

Educational Products Information Exchange Institute (EPIE)
P.O. Box 620
Stony Brook, NY 11790

An educational consumer advocacy group which, in conjunction with the Microcomputer Resource Center at Columbia University Teachers College, publishes detailed critical reviews of commercially available educational software.

Florida Center for Instructional Computing
College of Education
University of South Florida
Tampa, FL 33620

Funded in part through the Florida State Department of Education, the FCIC serves Florida educators by compiling software reviews and maintaining an educational software index.

Helping Schools and Community Colleges to Choose Microcomputer Courseware
c/o Dr. Vicki Blum Cohen
Microcomputer Resource Center
Box 18, Teachers College
Columbia University
New York, NY 10027

This project is producing detailed reviews of courseware in math, science, and communication skills designed for elementary through community college levels. EPIE publishes the reviews periodically.

Instructional Materials Division
Department of Education
State of New Mexico
Santa Fe, NM 87501-2786

The Instructional Materials Division reviews educational software and publishes a list of state adopted software.

Materials Review and Evaluation Center
North Carolina Dept. of Public Instruction
Raleigh, NC 27611

Reviews educational software and publishes a list of highly rated pieces, "Advisory List of Instructional Media," which is sent to all 50 state departments of education in the U.S. and to schools throughout North Carolina.

Michigan Association for Computer Users in Learning (MACUL)
33500 Van Born Road
Wayne, MI 48184

Collects and reviews public domain educational software for the Apple. Publishes revi[illegible] of educational programming.

Micro Co-op
P.O. Box 432
West Chicago, IL 60185

A software cooperative that distributes Apple and Atari software to members at reduced rates. Its bimonthly newsletter describes and compares programs.

Microcomputer Education Applications Network (MEAN)
256 North Washington Street
Falls Church, VA 22046

Aids educators in developing and selling software. Publishes a quarterly newsletter which contains information on software services.

Microcomputer Resource Center
Teachers College, Columbia University
525 W. 121st Street
New York, NY 10027

Maintains a collection of hardware, software, journals, and books, which educators are welcome to use. Runs seminars and workshops on microcomputer applications in the schools.

Microcomputer Software and Information for Teachers (MicroSIFT)
Northwest Regional Educational Laboratory
300 SW 6th Avenue
Portland, OR 97204

Disseminates descriptive and evaluative information about educational software. Its reviews are available through state and local education agencies, various periodicals, and the RICE database of the Bibliographic Retreival Services Computerized information system.

National Council of Teachers of Mathematics (NCTM)
1906 Association Drive
Reston, VA 20091

Publishes software reviews in its various periodicals. Developed *Guidelines for Evaluating Computerized Instructional Mater[illegible]*

SOFTSWAP
c/o Ann Lathrop
San Mateo County Office of Education
333 Main Street
Redwood City, CA 94063

Collects, evaluates, and modifies educational programs, and makes them available free of charge to educators who copy them at the center. Operates a software exchange which allows any educator who contributes a program to request one in exchange. Sells public domain software at low cost.

Technical Education Research Centers, Inc. (TERC)
8 Eliot Street
Cambridge, MA 02138

Conducts workshops and provides information services relating to educational computing. A software review service is in the planning stage.

Periodicals And Reports Devoted To Software Reviews

Computer Software: A Manual for Teachers
Teacher Center for Montana
215 S. 6th Street West
Missoula, MT 59801

A collection of reviews of software in Language Arts, Science, Soc[illegible] and [illegible], etc

Courseware Report Card
150 West Carob Street
Compton, CA 90220

5 issues per year. Separate elementary and secondary editions review educational software for Apple, Radio Shack, Atari, Commodore, and Texas Instruments.

Digest of Software Reviews: Education
1341 Bulldog Lane, Suite C
Fresno, CA 93710

Four issues per year; describes 50 programs per issue.

Dvorak's Software Review
704 Solano Avenue
Albany, CA 94706

Eight issues per year.

Journal of Courseware Review
The Apple Education Foundation
20525 Mariani Avenue
Cupertino, CA 95014

5 issues per year. Contains signed critical reviews of commercial educational programs for the Apple II.

MACUL Journal
c/o Larry Smith
Wayne County ISD
P.O. Box 807
Wayne, MI 48184

Occasional. The Winter 1981 issue was devoted to software reviews. Focuses on Apple II software.

Microcomputer Courseware/Microprocessor Games,
EPIE Materials Report 98/99m.
from: EPIE Institute
P.O. Box 620
Stony Brook, NY 11790

Includes reviews of six commercial educational software packages.

Microcomputers in Education
Queue, Inc.
5 Chapel Hill Drive
Fairfield, CT 06432

Monthly. Describes new educational programs, reviews software, and summarizes software reviews from other sources.

MICRO-SCOPE
JEM Research
Discovery Park
University of Victoria
P.O. Box 1700
Victoria, BC V8W 2Y2 CANADA

Monthly.

Peelings II
P.O. Box 188
Las Cruces, NM 88004

Nine issues year year. Describes and evaluates commercially available software for the Apple II.

Pipeline
P.O. Box 388
Iowa City, IA 52244

Twice yearly. Describes software reviewed and tested by the Conduit Clearinghouse.

PURSER'S MAGAZINE
P.O. Box 466
El Dorado, CA 95623

Quarterly. Reviews software for TRS-80, Apple, and Atari.

School Microware Reviews
Dresden Associates
P.O. Box 246
Dresden, ME 04342

Semi-annually. Includes detailed teacher evaluations of educational programs for the Apple, Atari, PET, and TRS-80.

Software Review
Meckler Publishing
520 Riverside Avenue
Westport, CT 06880

Quarterly. Reviews software designed for school and library use.

TALMIS Courseware Ratings
115 North Oak Park Avenue
Oak Park, IL 60301

Contains teacher produced evaluations and ratings for commercially published educational software.

Periodicals Containing Software Reviews

AEDS Monitor
1201 16th Street, N.W.
Washington, DC 20036

Four issues per year, with an average of two reviews per issue.

Applesauce
P.O. Box 598
Venice, CA 90291

Six issues per year. Contains reviews of Apple software.

Arithmetic Teacher
National Council of Teachers of Mathematics
1906 Association Drive
Reston, VA 22091

Nine issues yearly.

BYTE
70 Main Street
Peterborough, NH 03458

Monthly. Includes detailed discussions of new software.

Call — A.P.P.L.E.
Apple Puget Sound Program
Library Exchange
304 Main Avenue S., Suite 300
Renton, WA 98055

Monthly. Contains Apple software reviews.

Classroom Computer News
Intentional Educations, Inc.
341 Mt. Auburn Street
Watertown, MA 02172

Six issues yearly. Includes a regular review section focusing on educational software.

Compute!
515 Abbott Drive
Broomall, PA 19008

Monthly. Frequently features educational programs.

The Computing Teacher
Department of Computer and Information Science
University of Oregon
Eugene, OR 97403

Nine issues per year. Regularly includes reviews of educational software.

Creative Computing
P.O. Box 789-M
Morristown, NJ 07960

Monthly. Occasional special educational issues focus on software for school use.

C.U.E. Newsletter
P.O. Box 18457
San Jose, CA 95158

Six issues per year. Regularly reviews educational software.

Educational Computer Magazine
P.O. Box 535
Cupertino, CA 95015

Bimonthly. Includes software reviews in every issue.

Educational Technology
140 Sylvan Avenue
Englewood Cliffs, NJ 07632

Monthly. Regularly reviews commercially available educational software.

The Electronic Classroom
150 West Carob Street
Compton, CA 90220

Monthly. Contains detailed evaluations of software designed for classroom use.

Electronic Education
Electronic Communications, Inc.
Suite 220
1311 Executive Center Drive
Tallahassee, FL 32301

Ten issues yearly. Each issue contains a detailed review of a computer system or instructional package.

Electronic Learning
Scholastic Inc.
902 Sylvan Avenue
P.O. Box 2001
Englewood Cliffs, NJ 07632

Bimonthly. Each issue contains software reviews, and discussions of educational applications of microcomputers.

Hardcore Computing
14404 East D Street
Tacoma, WA 98445

Quarterly. Contains program listings, reviews and discussions of issues relating to software.

InfoWorld
375 Cochituate Road
P.O. Box 880
Framingham, MA 01701

Weekly. Geared toward the beginner. Contains detailed product reviews.

Interface Age
16704 Marquardt Avenue
Cerritos, CA 90701

Monthly.

Journal of Computers in Mathematics and Science Teaching
Association for Computers in Mathematics and Science Teaching
P.O. Box 4455
Austin, TX 78765

Quarterly. Contains software listings, reviews, and discussions of computer use in instruction.

Journal of Learning Disabilities
1331 E. Thunderhead Drive
Tucson, AZ 85718

Quarterly. Publishes reviews of software designed for handicapped populations.

Kilobaud Microcomputing
80 Pine Street
Peterborough, NH 03458

Monthly.

Mathematics Teacher
National Council of Teachers of Mathematics
1906 Association Drive
Reston, VA 22091

Nine issues/year. Emphasizes practical aids for mathematics education in secondary and junior college level. Includes software reviews.

80 Microcomputing
80 Pine Street
Peterborough, NH 03458

Monthly. Concerned with TRS 80 information only.

Microcomputer Digest
103 Bridge Avenue
Bay Head, NJ 08742

Monthly (except August). Information and comparisons of hardware and software for teachers and educational administrators.

Nibble
P.O. Box 325
Lincoln, MA 01773

Eight issues/year. Program listings, tips, and software reviews.

Personal Computing
P.O. Box 1408
Riverton, NJ 08077

Monthly.

Popular Computing
70 Main Street
Peterborough, NH 03458

Monthly.

T.H.E. Journal
P.O. Box 992
Acton, MA 01720

Bimonthly. Focuses on educational application of technology. Contains reviews of software, projects and publications.

Window
469 Pleasant Street
Watertown, MA 02172

A magazine on disk, issued 5 times per year. Allows users to try out software.

Computer Accessible Databases

The BRS and Dialog online bibliographic search services provide access to databases containing software information.

Databases Accessible through BRS

SPIF (School Practices Information File)

Contains more than 1,500 descriptions of educational software. Includes the MARCK and MicroSIFT software catalogs.

RICE (Resources in Computer Education)

Includes descriptions of educational software and information concerning software producers.

Bibliographic Retrieval Services
1200 Route 7
Latham, NY 12100

Databases Accessible through Dialog

International Software Databases

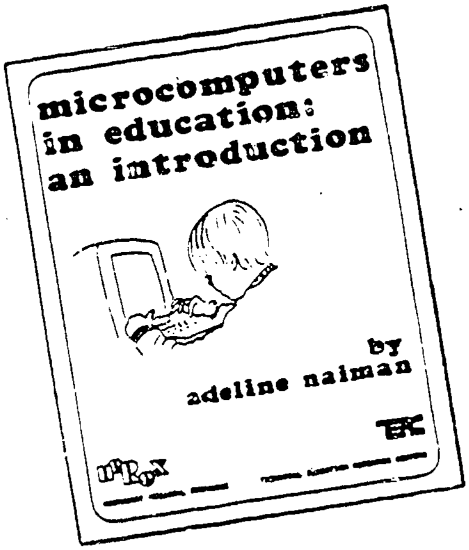
Contains descriptions of all types of software, contributed by vendors throughout the world. Includes references to independent reviews.

Microcomputer Index

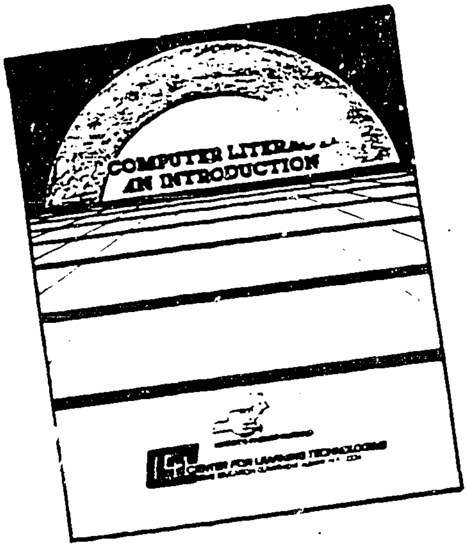
Indexes more than 25 English language microcomputer periodicals.

Dialog Information Retrieval Services
3460 Hillview Avenue
Palo Alto, CA 94304

RESOURCE MATERIALS FROM THE EXCHANGE



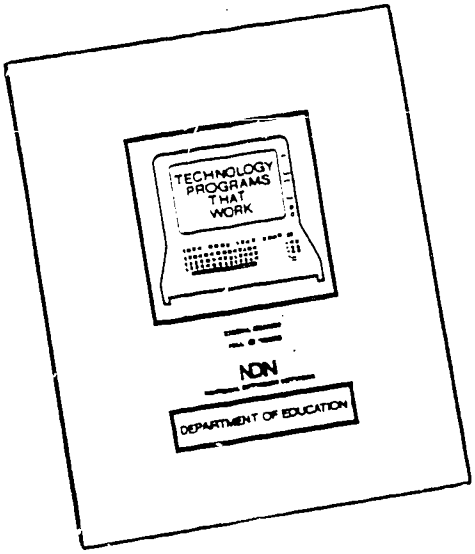
This resource book resulted from the cooperative work of key education professionals in the seven state region of the Northeast, the staff and financial support of the Exchange and the contractual services of Technical Education Research Centers in Cambridge, Massachusetts. As a service to the region NEREX has disseminated at no charge over 3000 copies of this resource. The seven state departments of education in our region helped distribute the book to teachers and other educators who are promoting the effective use of technology in education. As a personal resource and as an in-service text book Microcomputers in Education: An Introduction is helping educators plan ways to successfully use computers as instructional aids. Since the demand for this book has been so high we have printed additional copies for sale. You can order additional copies with the form on the next page.


COMPUTER LITERACY: AN INTRODUCTION
CENTER FOR LEARNING TECHNOLOGIES

This resource book was produced by the New York State Education Department's Center For Learning Technologies under a State Priority Grant from the Northeast Regional Exchange, Inc. New York has disseminated over 10,000 copies of this book within their own state and NEREX has disseminated over 1,000 copies to other states in the region.

Computer Literacy: An Introduction, provides a national, state, and local perspective on computer literacy. Featured are sections describing results of a national survey of computer experts, state and local activities, reprints of key journal articles depicting varying views, materials about literacy for school personnel and the public, materials about computer literacy for students and annotated bibliographies.

Since the demand for this resource book has been so high we have printed additional copies for sale. You can order additional copies with the form on the next page.



Technology Programs That Work is an National Diffusion Network catalog of exemplary educational programs validated on the basis of their effective use of technology. This catalog was developed and printed at the private expense of The Northeast Regional Exchange, Inc., a not-for-profit educational service organization.

The catalog contains descriptions of nineteen National Diffusion Network Technology Programs including the four new NDN Technology Lighthouse Projects. Please use the form on the next page to order additional copies.

northeast regional exchange

101 MILL ROAD, CHELMSFORD, MA 01824 • 617 / 256-3987

Microcomputers In Education: An Introduction

	Quantity	Amount
1 Copy @ $6.00		
10 or More Copies @ $4.50		
100 or More Copies @ $4.00		

Computer Literacy: An Introduction

	Quantity	Amount
1 Copy @ $6.00		
10 or More Copies @ $5.00		
100 or More Copies @ $4.50		

Technology Programs That Work

	Quantity	Amount
1 Copy @ $3.50		
10 or More Copies @ $3.00		
100 or More Copies @ $2.50		

TOTAL AMOUNT	

All prices include shipping costs.
All orders less than $25.00 must be prepaid. Make checks payable to NEREX, Inc.

Name: __________________________

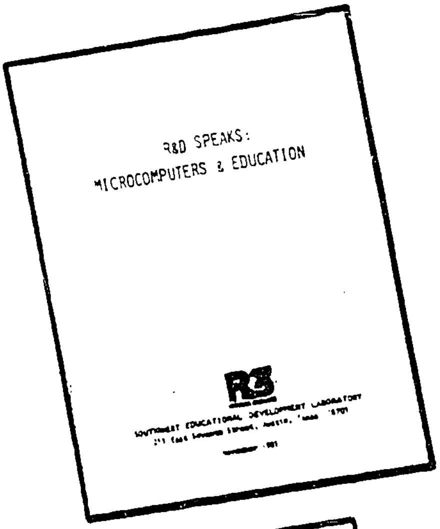
Organization: __________________________

Address: __________________________

____

Send to:

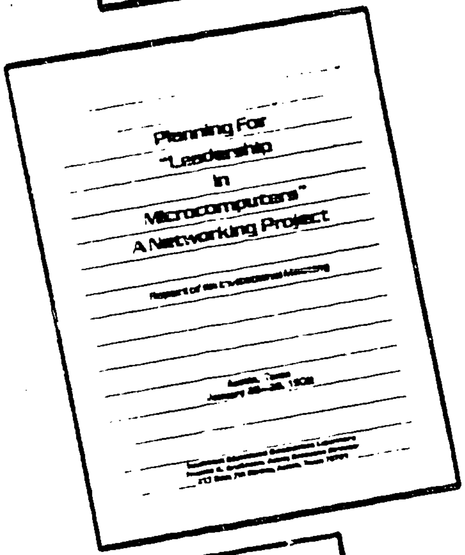
NORTHEAST REGIONAL EXCHANGE, INC.
101 MILL ROAD
CHELMSFORD, MA 01824



R&D SPEAKS:
MICROCOMPUTERS & EDUCATION

SOUTHWEST EDUCATIONAL DEVELOPMENT LABORATORY
211 East Seventh Street, Austin, Texas 78701

R&D Speaks: Microcomputers and Education is the proceedings from an invitational conference of state education agency and local school district personnel held at SEDL in October, 1981. A service of the Regional Exchange Project, the conference was designed to provide participants with a forum for discussing major issues such as the role of state education agencies, teacher training in microcomputer use, and evaluation of software. The book includes a discussion of major issues to consider in microcomputer use; computing competencies for teachers; EPIE and MicroSIFT approaches to software evaluation and information on approaches to microcomputer use in six school districts and four state education agencies (Arkansas, Louisiana, Oklahoma, and Texas). $6.00

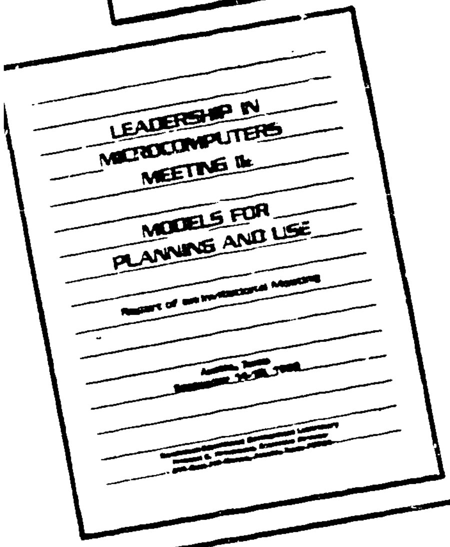


Planning For
"Leadership
in
Microcomputers"
A Networking Project

Report of an Invitational Meeting

Austin, Texas

Planning for "Leadership in Microcomputers" A Networking Project is the proceedings from a conference of state education agency personnel held at SEDL in January, 1982. A service of the Regional Planning and Service Project, the conference was designed to assist SEAs in moving quickly into a leadership role in computerized instructional systems by forming a network of state agencies computer companies, and SEDL. This book contains discussions of the following topics: What Computer Related Competencies are Needed by SEA Staff Members? How Can Project Activities be Most Effectively Managed in Each State? What are the Optimum Positions and Actions for SEAs to Take to Achieve Leadership in Microcomputer Use for Education? plus summaries of key presentations by Norman Bell, Jim Dugan, and Jim Poirot. $5.00

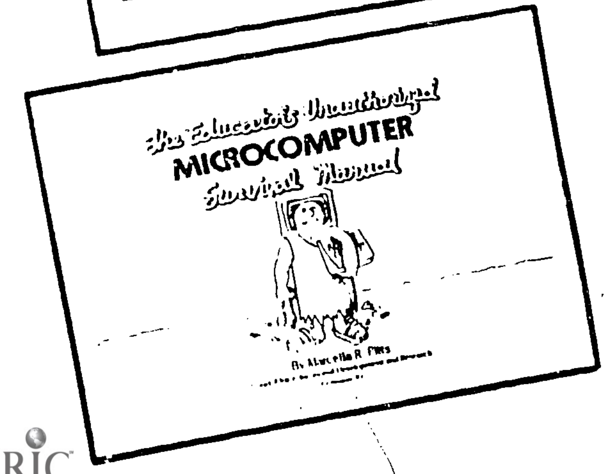


LEADERSHIP IN
MICROCOMPUTERS
MEETING II:
MODELS FOR
PLANNING AND USE

Report of an Invitational Meeting

Austin, Texas

Leadership in Microcomputers Meeting II: Models for Planning and Use is the report of an invitational meeting of state education agency personnel held at SEDL in September, 1982. A service of the Regional Planning and Service Project, the meeting was a follow up to the January 1982 "Leadership in Microcomputers" meeting described above. This book contains reports from six SEAs (Arkansas, Louisiana, Mississippi, New Mexico, Oklahoma, and Texas); discussions of state and local educational uses of technology (including MECC, the California SEA, the Lexington MA schools, and the Lyons Township, IL schools); and appendices. $5.00



The Educator's Unauthorized
MICROCOMPUTER
Survival Manual

By Marcella R. Fitts

The Educator's Unauthorized Microcomputer Survival Manual was produced in 1982 by the Council for Educational Development and Research, a non-profit association of educational research organizations to which SEDL belongs. The manual is designed for anyone facing decisions about where, how, and whether to use microcomputers for instruction. The manual's view of technology is "different from what you'll hear from technology enthusiasts in government and from computer manufacturers backing legislation to increase the number of microcomputers in the schools." The book contains such discussions as "survival tactics," computer jargon, deciding how to use microcomputers, and selecting software. $2.00