DOCUMENT RESUME

ED 232 063 CE 036 762

AUTHOR	Stone, Antonia
TITLE	Microcomputer Software for Adult Vocational Education: Guidelines for Evaluation. Information Series No. 261.
INSTITUTION	ERIC Clearinghouse on Adult, Career, and Vocational Education, Columbus, Ohio.
SPONS AGENCY	National Inst. of Education (ED), Washington, DC.
PUB DATE	83
CONTRACT	400-81-0035
NOTE	35p.
AVAILABLE FROM	National Center Publications, National Center for Research in Vocational Education, 1960 Kenny Road, Columbus, OH 43210 (Order No. IN261).
PUB TYPE	Information Analyses - ERIC Information Analysis Products (071) -- Guides - Non-Classroom Use (055)
EDRS PRICE	MF01/PC02 Plus Postage.
DESCRIPTORS	Adult Education; *Adult Vocational Education; *Computer Assisted Instruction; Computer Managed Instruction; *Computer Programs; *Evaluation Criteria; Guidelines; Individualized Instruction; *Microcomputers; Programing

ABSTRACT

This guide addresses the applicability of microcomputers to adult vocational training and presents guidelines for creating and evaluating software that specifically addresses adult learning needs. The author first discusses the appropriateness of microcomputers as educational and prevocational tools. She highlights those qualities that are the computer's special contributions to the learning process, such as individualization, impartiality, and capacity for routine, and then explains how these features can eliminate barriers to adult learning. Categories of educational software follow, illustrating the use of the computer (1) as an instructional medium, through drill and practice, tutorials, and demonstrations; (2) as a modeling device, through games and simulations; and (3) as a special-purpose, general-purpose, or tool-making (i.e., programming and authoring languages) tool. The remainder of the document outlines the guidelines for software development and assessment, divided into five areas: (1) learning objectives and task analysis, (2) appropriate use of technology, (3) pedagogical considerations, (4) management considerations, and (5) necessity, content, and format for accompanying textual material. In concluding, the author notes the lack of available software that meets these guidelines and suggests using general-purpose software, exploring public-domain programs, investing in commercially available open-ended packages, and encouraging corporations to share their training software. (SK)

ED232063

Information Series No. 263

MICROCOMPUTER SOFTWARE FOR ADULT VOCATIONAL EDUCATION: GUIDELINES FOR EVALUATION

Antonia Stone
Playing to Win, Inc.
New York, New York

The ERIC Clearinghouse on Adult, Career, and Vocational Education
The National Center for Research in Vocational Education
The Ohio State University
1960 Kenny Road
Columbus, Ohio 43210

1983

CE036762

THE NATIONAL CENTER MISSION STATEMENT

The National Center for Research in Vocational Education's mission is to increase the ability of diverse agencies, institutions, and organizations to solve educational problems relating to individual career planning, preparation, and progression. The National Center fulfills its mission by:

- Generating knowledge through research
- Developing educational programs and products
- Evaluating individual program needs and outcomes
- Providing information for national planning and policy
- Installing educational programs and products
- Operating information systems and services
- Conducting leadership development and training programs

For further information contact:

Program Information Office
National Center for Research in Vocational Education
The Ohio State University
1960 Kenny Road
Columbus, Ohio 43210

Telephone: (614) 486-3655 or (800) 848-4815
Cable: CTVOCEDOSU/Columbus, Ohio
Telex: 8104821894

FUNDING INFORMATION

Project Title:	ERIC Clearinghouse on Adult, Career, and Vocational Education
Contract Number:	NIE-C-400-81-0035
Educational Act Under Which the Funds Were Administered:	41 USC 252 (15) and PL 92-318
Source of Contact:	U.S. Department of Education National Institute of Education Washington, D.C. 20208
Contractor:	The National Center for Research in Vocational Education The Ohio State University Columbus, Ohio 43210
Executive Director:	Robert E. Taylor
Project Director:	Juliet V. Miller
Disclaimer:	This publication was prepared pursuant to a contract with the National Institute of Education, U.S. Department of Education. Contractors undertaking such projects under government sponsorship are encouraged to express their judgment in professional and technical matters freely. Points of view or opinions do not, therefore, necessarily represent official National Institute of Education position or policy.
Discrimination Prohibited:	Title VI of the Civil Rights Act of 1964 states: "No person in the United States shall, on the ground of race, color, or national origin, be excluded from participation in, be denied the benefits of, or be subjected to discrimination under any program or activity receiving federal assistance." Title IX of the Education Amendments of 1972 states: "No person in the United States shall, on the basis of sex, be excluded from participation in, be denied the benefits of, or be subjected to discrimination under any education program or activity receiving federal assistance." The ERIC Clearinghouse project, like every program or activity receiving financial assistance from the U.S. Department of Education, must comply with these laws.

This publication was prepared with funding from the National Institute of Education, U.S. Department of Education, under Contract No. NIE-C-400-81-0025. The opinions expressed in this report do not necessarily reflect the position or policies of NIE or the Department of Education.

CONTENTS

FOREWORD

The Educational Resources Information Center Clearinghouse on Adult, Career, and Vocational Education (ERIC/ACVE) is one of sixteen clearinghouses in a nationwide system that is funded by the National Institute of Education. One of the functions of the Clearinghouse is to interpret the literature that is entered into the ERIC database. This paper should be of particular interest to adult vocational education practitioners and decision makers.

The profession is indebted to Antonia Stone for her scholarship in the preparation of this paper. Ms. Stone is Founder and Executive Director of PLAYING TO WIN, INCORPORATED, a nonprofit organization dedicated to promoting educational computer use for minorities, inmates of correctional institutions, juvenile delinquents, and other people who are socially handicapped. She has led workshops and delivered keynote speeches at a number of national and regional conferences on the topic of educational software. Prior to founding PLAYING TO WIN, Ms. Stone was a mathematics educator in New York City.

Recognition is also due to Vicki Blum Cohen, International Software Systems; to Judith Rodenstein, the Vocational Studies Center, the University of Wisconsin; and to Constance Faddis and Gale Zahniser, the National Center for Research in Vocational Education, for their critical review of the manuscript prior to its final revision and publication. Susan Imel, Assistant Director at the ERIC Clearinghouse on Adult, Career, and Vocational Education, coordinated the publication's development. She was assisted by Sandra Kerka and Judith O. Wagner. The manuscript was typed by Linda Adams, with Brenda Hemming and Janet Ray serving as word processor operators. Editing was performed by Rod Spain of the National Center's Editorial Services.

Robert E. Taylor
Executive Director
The National Center for Research
in Vocational Education

PREFACE

Creating criteria for evaluation and development of microcomputer software in any educational arena is an awesome task. While I have attempted in this paper a coalescence of various published efforts with my own views, I am hopeful that readers will realize that, in the end, all choices must be their own.

By way of preface, I offer this statement from an issue of the Mathematics Teacher:

> The classification of an item of software as good or poor (useful or useless) is, in the final analysis, a professional judgment. It is our privilege to make such judgments and our fate to live with the results. All the world's questions, checklists, consultants, and research are means, not ends, in this decision-making process. (Kansky, Heck, and Johnson 1981, p. 604)

EXECUTIVE SUMMARY

This guide addresses the applicability of microcomputers to adult vocational training and presents guidelines for creating and evaluating software that specifically addresses adult learning needs. The author first discusses the appropriateness of microcomputers as educational and prevocational tools. She highlights those qualities that are the computer's special contributions to the learning process, such as individualization, impartiality, and capacity for routine, and then explains how these features can eliminate barriers to adult learning. Categories of educational software follow, illustrating the use of the computer (1) as an instructional medium, through drill and practice, tutorials, and demonstrations; (2) as a modeling device, through games and simulations; and (3) as a special-purpose, general-purpose, or tool-making (i.e., programming and authoring languages) tool. The remainder of the document outlines the guidelines for software development and assessment, divided into five areas: (1) learning objectives and task analysis, (2) appropriate use of technology, (3) pedagogical considerations, (4) management considerations, and (5) necessity, content, and format for accompanying textual material. In concluding, the author notes the lack of available software that meets these guidelines and suggests using general-purpose software, exploring public-domain programs, investing in commercially available open-ended packages, and encouraging corporations to share their training software.

Information on microcomputer software evaluation can be found in the ERIC system under the following descriptors: *Microcomputers; *Computer Programs; *Adult Vocational Education; Adult Education; Guidelines; *Computer Assisted Instruction; *Evaluation Criteria; Individualized Instruction; Computer Managed Instruction; Programing. Asterisks indicate descriptors having particular relevance.

INTRODUCTION

Educators are not known for their speed in adjusting to new technologies, and the record shows that those working with the fledgling microcomputer technology are no exception. Considerable efforts, however, have recently been made to introduce courses in computer awareness, computer literacy, and even programming into many of our public schools (U.S. Department of Education 1982). Companies devoted to the creation of educational programs for microcomputers--known as software--are proliferating, and since 1980, government agencies and foundations have funded intensive studies designed to assist educators in creating and evaluating educational software. Unfortunately, the creation of microcomputer software specifically addressing the problems of adult vocational education has been largely neglected.

Software does exist, however, that can be used or could be adapted for use in some aspects of adult vocational training. Because the computer was originally perceived by many to be primarily a business tool, some commercial software exists for almost every model of microcomputer to provide drill in typing, data entry, accounting, and other office skills. Furthermore, a number of large companies as well as the United States armed forces have developed specific task-oriented computer programs (Rahmlow 1979). Their software is, however, generally designed for those costly mainframe or minicomputers in use by the military service or corporations and is not compatible with microcomputers.* For the most part, it is designed to teach a particular technical skill such as assembling a Hoover vacuum cleaner, troubleshooting a Ford Pinto engine, maintaining the M-1 rifle, installing an American Bell push-button phone, or issuing a New York Life Insurance Company policy. Even this brief catalog is sufficient to suggest that creation of microcomputer software to address each separate vocational task would be a monumental problem, particularly in those industrial areas where technology is changing so quickly. Nevertheless, the unique capabilities of the computer serve so well the need for simulated experience that the usefulness of acquiring or generating task-specific software for adult vocational training must be considered.

In major areas of vocational training, albeit more general ones, microcomputers can be an important tool: writing skills, a paramount consideration in most vocations; reading and math; and, increasingly important, the skills involved in using the computer itself. In these areas, both commercially available and public domain software does exist.

*Software produced for a specific make and model of computer cannot, in the vast majority of cases, be used on any other model or make.

The questions, then, are: (1) is the microcomputer technology appropriate and necessary to the needs of community-based organizations and other agencies addressing the vocational concerns of adults; and (2) if so, what guidelines should be used for developing and evaluating software aimed at this broad target market? It must be kept in mind in considering education for vocational and career skills that adults in need of and seeking such training will range from those adults with a reasonably good basic education to those who are among the millions of known functionally illiterate in the country.*

---

*Estimates of numbers of functionally illiterate adults in this country range from 50 million to 65 million depending on data source and definition (Ulmer and Doriand 1981).

APPROPRIATENESS AND NECESSITY

This section discusses why the microcomputer is an appropriate educational tool for use with adults. It also explores why adults should become familiar with and learn to use the microcomputer.

The Microcomputer as an Educational Tool

Because of its potential for individualizing and personalizing instruction, the microcomputer is a promising educational tool. Some adult educators may oppose the use of microcomputers on philosophical grounds, feeling that its use detracts from interpersonal interaction in the learning process. The reverse, however, may actually be true. Use of a microcomputer may enhance personal interaction between students and the instructor since it can free instructors from basic instructional tasks and allow them to concentrate on the students' more complex and personal concerns (Garrison 1982; Meierhenry 1982).

As a learning tool in specific curricular areas, the computer has unique qualities. It has no feelings; it doesn't criticize, sympathize, become angry, interpret, or react except in the ways in which it has been programmed. This is at once a limitation and an asset. While the computer cannot sense a student's lack of understanding nor adjust its explanations to nuances of a student's interpretation of material, it can pose the very same question ten, fifteen, even a thousand times without any trace of impatience at the lack of a correct response. All students receive the same machine reaction whether they are young, old, black, white, female, male, quick, slow, rich, or poor. This admittedly mechanistic quality of the computer has the advantage of being totally unprejudiced and nonjudgmental. It can eliminate the anxiety often present in conventional classrooms, particularly with adult learners who have had negative experiences with education. Students using a microcomputer do not have to waste energies over being "liked" (feeling approval) or "disliked" (feeling humiliated) and can, therefore, concentrate more completely on solving the problem or digesting the material presented.

On the other hand, some adults need the reassurance that can only be provided by human contact. Educators using microcomputers to provide vocational training for adults should make sure resource persons are available to assist students and to offer guidance and support.

While a computer makes no emotional judgments about the operator, the capacity of the machine for storage and retrieval of information does allow for individualization of student use so that individuals are not only interacting with material at their own level and speed but are also receiving instant individualized responses. Individualized instruction is a very important feature when working with adults. Adults have a variety of learning needs; they learn at varying speeds; in addition they must schedule educational programs around a number of other activities. Use of the microcomputer allows adults flexibility in pursuing educational goals (Garrison 1982).

Experience has shown that the interaction between students and the microcomputer, enhanced by the machine's capabilities for using color, sound, graphics, and animation, keeps students involved and motivated to a remarkable degree and greatly extends their attention span. This motivation will be especially important for adults who have a passive and negative approach to learning due to past failures (Ayers 1980).

In addition to providing motivation, the computer is capable of animating the construction of an automobile engine, designing schematics and graphs, or presenting routine drill and practice in the guise of a game. It not only can make many routine vocational learning tasks palatable but can also put other, less routine and perhaps heretofore inaccessible, learning activities within the reach of adult students. Through the use of a microcomputer, for example, adults can engage in increasingly complex problem-solving activities (Ayers 1980).

In virtually any vocation or career, reading and writing are essential elements. Many adults who lack basic skills have become skilled in masking their difficulties with reading and writing. The microcomputer, however, does not have to be fooled because, as a learning tool, it is private, patient, consistent, and as noted earlier, nonjudgmental (Judd 1982).

The computer has proven successful in helping adults to improve their reading skills. In one study conducted with adult basic education (ABE) students in Florida, participants gained on the average almost three quarters of a grade level for every twenty hours of computer-assisted instruction (Judd 1982).

As a substitute for the printed page, a computer screen offers no particular advantages to the reader. When used simply for displaying text, in fact, the printed page has distinct advantages over the screen, most notably clearer type (resolution) and portability. As a tool for <u>creating</u> text, however, the computer screen, coupled with word-processing software, can vastly simplify and speed up the tasks of editing. Rearranging sentences or whole paragraphs, altering a word or phrase, and correcting spelling and punctuation are all faster and far easier to cope with on the computer than with a typewriter or pencil. By reducing this drudgery in writing, and particularly in rewriting, the challenges of expressing oneself "on paper" become less intimidating. Adults can learn writing skills with less stress and more effectiveness on a computer.

The Microcomputer as a Prevocational Tool

Any exposure adults receive to microcomputers in their courses of study can be a great benefit in itself since computer experience is becoming a requirement for much entry-level employment. In the huge white-collar field of office work, all but the most menial jobs now, or very soon, will require the ability to use computers at some level of competence. Even the entry-level file clerk jobs are affected as "files" become "databases." By 1990, 40 million people will be needed to operate computers in the United States, and by 1985 eight of ten adults will need to use a computer to function in their jobs (Martin 1982).

By working with computers in their day-to-day training, adults will develop competence in handling menu-driven programs and the basic vocabulary of the technology. They will come to accept computer work as routine. The apprehension that many older students may have about this new technology will be reduced or disappear. A positive attitude will be fostered about the computer's capabilities and about the adults' abilities to participate in a computerized world. Additionally, some adult students will decide to pursue computer technology as a career.

If the foregoing judgments are accepted, it becomes an inescapable conclusion that incorporating the use of computers with appropriate, well-designed software can dramatically and dynamically enhance a training program. Such use can also provide a practical introduction to computer technology as a vocation.

CATEGORIES OF SOFTWARE

Once the fundamental decision has been reached to provide at least some training for adults on microcomputers, the real problem for vocational education organizations begins. What is the most suitable hardware? Which software packages should be used? How much will they cost? Where can they be obtained? How much teacher training will be needed before teachers can integrate computer technology effectively into the curriculum?

The choice of software is likely to determine the answers to the other questions, so investigation of and planning for software should be the first consideration. Any discussion of criteria for software development or selection must be prefaced by some description of the types or categories of educational software. Various lists of such categories have been proposed by educators. One such list, widely used, was developed by Richard Dennis (1979) of the University of Illinois at Champaign/Urbana. Dennis divides educational software into eight modes as follows:

- Drill and Practice
- Instructional Gaming
- Instructional Management (Computer-managed Instruction)
- Instructional Support
- Problem Solving and Research
- Simulation
- Test Construction and Analysis
- Tutorial Instruction (Computer-assisted Instruction)

Related to Dennis' modes, but perhaps better suited to the purpose of categorizing software for vocational education of adults, are the three categories enumerated by Henry Olds (1983) in an introductory chapter from Evaluation of Educational Software: A Guide to the Guides:

> 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 tool with which the user performs some task. (p. 5)

Olds goes on to provide subheadings in each of these categorizations. Figure 1 depicts an adaptation of Olds' categories and subheadings.

Category I -- Computer-assisted Instruction

- Drill and Practice
- Tutorials
- Demonstrations

Category II -- Modeling Devices

- Games
- Simulations

Category III -- Computer as Tool

- Special Purpose Tool
- General Purpose Tool
- Tool-making Tool

Figure 1. Software Categories for Adult Vocational Education (Adopted from Olds 1983)

Computer as Instructional Medium

Under the category of computer as an instructional medium, Olds cites drill and practice as one and tutorials as another instructional medium. Although not included under this heading by Olds, demonstrations might well be a third. These categories are commonly referred to as computer-assisted instruction (CAI).

Drill and Practice

Drill and practice programs are designed to increase speed and accuracy of performance in some well-defined skill area in which the student has already received instruction. Practicing multiplication problems, using spelling and vocabulary drills, and typing exercises are examples. Such software, appropriately enhanced by sound and color graphics, can make any necessary rote learning more attractive.

Tutorials

Tutorials refer to those programs that purport to teach directly. A practical tutorial requires extensive hardware memory or storage capability since it must allow for both slow and fast learners as well as for a variety of instructional opportunities. The technique of allowing a student's response to guide his or her progress to the appropriate level is called "branching." The inclusion of branching is the mark of a sophisticated tutorial (but not necessarily a good or accurate one). Olds' (1983) caution that tutorials "work best when a high level of motivation can be assumed on the part of the user" (p. 5) may be a tactful way of cautioning his readers that much current software falling in this category is extremely boring, sure evidence that the microcomputer is still a long way from being a human teacher substitute.

Demonstrations

Demonstrations are programs designed as illustrations. They require little interaction with the machine other than the selection of items to be demonstrated. Examples might include the generation of graphs representing different equations (or for comparison of information), schematics of the passage of fuel through an engine (or food through the human body), or illustration of the genetic traits in successive generations of fruit flies. The lack of student interaction in such programs makes them most suitable for class use by teachers who use animated supplements to chalkboard illustrations or by students who use the programs for review.

Computer as Modeling Device

Olds' second major software category, computer as modeling device, lists games and simulations. The essential difference between the two is that in games the situation into which the user is placed is artificial with an artificial goal; for example, a player must capture all his/her opponent's pieces, as in checkers. In simulations, on the other hand, a real-life problem is modeled. The player must evolve a winning or successful strategy to solve the problem but, of course, need not suffer the REAL consequences of erroneous judgments. Flight simulations have been used for years to train aircraft pilots; the 'crash" resulting from incorrect response to the simulated situation will not result in loss either of life or of expensive equipment. Simulations are being used more and more frequently by major corporations for training of personnel in the use of heavy or complex equipment. Simulations are especially appropriate for use with adults since adults respond well to learning situations that draw on real-life experience (Little 1981).

Software under both these subheadings challenges students to develop problem-solving strategies. It can even motivate the student to acquire specific information necessary to win the game or master the technique of the simulation. For these reasons commercial software producers are designing an increasing number of their products in these formats. Some programs that are marketed as games but, in fact, simply ask questions and reward a correct answer with points more properly belong in the drill and practice category. In one imaginative program ("Rocky's Boots," produced by the Learning Company), the student is guided to an understanding of logical circuitry design by adventuring through a maze of rooms in which objects may be picked up, put down, connected, or disconnected for the purpose of constructing a circuit to accomplish a specific task. Such a program combines the best elements of the technology to produce an innovative and absorbing approach to learning.

Computer as Tool

Olds' last general category, computer as tool, may well be the most important for vocational educators. It has been noted that the ability to use a computer as a tool is in itself a vocational skill. Software in this category provides opportunities for students to learn how to use computers--as well as how to use related skills.

Olds cites three types of software tools: (1) those for a specif[illegible] purpose such as the correction of spelling; (2) those with more gener[illegible]plication such as word processors that allow users to create, correct [illegible]t, and print text; spread-sheet programs* used for budgeting, forecas[illegible], and limited accounting procedures; and "utilities" that afford opportunities to add animated graphics or sound for a particular purpose; and (3) "tool-making tools" that are either computer languages with which the user creates his or her own software or secondary languages (called "authoring languages") that permit users to construct specific learning tools such as tests or drill and practice routines without having to master the intricacies of a full programming language.**

The virtues of using a word-processing program in adult vocational education have already been discussed. The same kind of argument could apply to the incorporation of a spread-sheet program in the curriculum. Spread-sheet programs have wide application. Not only do they remove the drudgery of successive calculation, but they provide the means with which to experiment with statistics, volumes, current flow, water displacement, stress, tolerances, and so forth. Constructing and using formulas become necessary and therefore more easily learned; the concept of "variable" becomes useful rather than threatening.

The programming language, LOGO, planned and designed under National Science Foundation auspices at the Massachusetts Institute of Technology by a team of educators and scientists directed by Seymour Papert, provides an apt example of Olds' last subcategory--tool-making tools. A version of LOGO is commercially available for almost every make of microcomputer, but other, more cogent reasons exist for using LOGO to introduce adult learners to computer programming:

- The language was designed for use by very young children and, thus, is appropriate for those adults whose education is minimal.

---

*A spread-sheet program utilizes matrixes (i.e., charts, tables) in which values change automatically as new values are added.

**The capacity to understand commands and execute instructions written in the computer language BASIC is built in to virtually every microcomputer.

- The commands used to instruct the computer (to program) result in instantaneous graphic displays so that users know immediately if their commands have elicited the expected response and, if this is not the case, are motivated to discover the error (debug the program).

- The language of LOGO is structured so that each user may increase LOGO's vocabulary by introducing and defining new words and commands.

This last feature provides a model of how all languages, human and computer, are built and grow and, thus, serves an additional educational role.

Programming instruction for adult students may have other benefits as well. In a study made at Yale University (Soloway, Lockhead, and Clement 1982) comparing the problem-solving ability of a group of students who had been introduced to computer programming with those of a group with no programming instruction, it was found that the problem-solving resources, the attack skills, and the willingness and ability to discover and correct errors were significantly higher among those who had had the programming experience. Problem solving is, after all, what any educational program is all about.

Application to Adult Learning Needs

The categories of software described previously can be used to meet a range of learning needs of adults enrolled in vocational training programs. While software in the computer-assisted instructional category may be most appropriate for adults with elementary learning needs, software programs in the modeling-devices and the computer-as-a-tool categories can address complex learning needs.

Many adults enrolled in vocational programs are underprepared learners with the following characteristics:

- They lack basic conceptual and computation skills.

- Their prior learning has been unsuccessful and ineffectual; therefore, they tend to approach learning with a low-risk-taking and self-protective attitude.

- They lack communication skills.

- They tend to depend on an authority figure in learning situations.

- They are not able to make judgments objectively.

- They have a passive and negative approach toward learning due to past failures (Ayers 1980).

Computer software in the computer-assisted instruction mode (i.e., drill and practice, tutorial, and demonstration) can provide useful and timely instructional solutions to many of these learning needs. Learners find the instruction to be beneficial and less threatening, and they value its self-paced and individualized nature. In addition, use of software in the CAI mode can help students develop such cognitive skills as recall of information, imitation, discovery of correct answers, simple calculation, and grasp of elementary ideas and patterns (Ayers 1980; Garrison 1982).

Software from the categories of modeling devices (games and simulations) and computer-as-a-tool address more complex learning needs and are appropriate for adults whose learning style is more autonomous and self-directed. Examples of cognitive skills developed by use of software in these categories include applying information to new situations, analyzing relationships, synthesizing elements into a new whole, recognizing unstated assumptions, and testing hypotheses (Ayers 1980).

Any attempt to describe software categories must include a caveat. One must remember that the state of the art of software development and educational computer use is in its infancy. Existing categories and new categories now being devised reflect only the experience of the first steps in what will be a great adventure: exploring the potential for creative learning with a tremendously powerful tool.

GUIDELINES FOR EVALUATION AND DEVELOPMENT

Guidelines for the creation or evaluation of any piece of work are necessarily personal. Just as a moviegoer may refuse to be a slave to one reviewer's opinion, so a potential evaluator or developer of educational software should become acquainted with a number of differing views regarding the applicable standards and try to arrive at a practical and personal synthesis.*

The following guidelines are divided into five areas of consideration:

I. Learning Objectives and Task Analysis

II. Appropriate Use of the Technology

III. Pedagogical Concerns

IV. Management Considerations

V. Necessity, Content, and Format for Accompanying Textual Material

Learning Objectives and Task Analysis

The two most important questions to be answered in either the development of software or the evaluation of an existing program are, "What educational objective is addressed?" and "What is the learning task of the student?" The more specific the objective, the easier it is either to evaluate or to plan a program. Once the objective and task have been identified, peripheral questions need to be asked, such as, "Is the computer appropriate for the accomplishment of this objective?" and "What category of computer software can provide the best vehicle for the achievement of the learner's task or goal?"

At this point it might be advisable to look at the ways in which the particular task or objective has been approached in traditional teaching environments and ask, "In what way will the use of a computer improve on established techniques for achieving this goal or produce effective alternatives to the traditional approach?"

For developers, a way of identifying specific teaching areas in which computer use could be constructive is to ask, "What elements of the particular

---

*See the bibliography for sources of other evaluation materials, forms, and guidelines. The previously cited Guide to the Guides (Olds 1983) is particularly recommended. Microcomputers in Education--Resource Lists, published by Technical Education Resource Center (1982), is also an excellent source of information.

discipline or subject matter have traditionally posed teaching problems and why?" If the problem is, for example, one of possible student injury or damage to expensive equipment caused by students' attempts to get experience in assembly or diagnosis, a computer simulation is suggested. If the problem is that students are not motivated to drill themselves in a required vocabulary or skill area, a lively computer game that demands accuracy and speed in the problem area may stimulate a larger number of students to acquire the requisite skill.

Appropriate Use of the Technology

How is the unique combination of capabilities of the computer for graphics and animation, sound and speech reproduction, interactivity and randomization employed? Attractive as these capabilities may be, each carries with it temptations for overuse or inappropriate use.

Graphics

The extent to which the graphics treatment is integrated with the subject matter may be difficult to judge but not hard to perceive when the graphics overpower the learning objective, becoming the focus of the user's attention. Creating graphics is such fun that the programmer can become infatuated with the sight of the images, little realizing that the student will become bored and impatient with purposeless repetition or long sequences of pictures that require no response. Any graphics display that impedes the student's progress through the program or that consumes time for nonlearning and nonmotivational purposes should be avoided. And of course graphics displays should not reward incorrect or inappropriate responses. An oft-cited example is a computer version of the familiar HANGMAN game in which students may deliberately try to misspell a word in order to view the dire, but visually more intriguing, consequences. At the other end of the spectrum is the program that neglects the computer's ability to model or picture, relying on word-for-word duplication of some textual material.

Some text is, of course, necessary in most programs. Care must be taken to ensure that such text, when presented on the computer screen, is clear and easy to read. Since regular computer characters are small and often fuzzy when reproduced on a screen, particularly if a television is used as a monitor, the design or use of special alphabetics may be required. Care must also be taken with the use of color to ensure sufficient contrast between characters and background. Green against blue may look lovely on a color monitor, but a user with only a black and white video display may see nothing at all.

Sound and Speech

If sound is a feature of the program, similar criteria apply. Does the sound serve a necessary function, or is it merely a distraction? Is the program just as effective with the sound turned off? Can you, in fact, turn it off? In a classroom with a number of machines operating, too many bleeps, bangs, wooshes, and fanfares played at too high a volume could create seriously annoying distractions for students and teachers alike.

Speech synthesis capability, not yet fully implemented on microcomputers, may nevertheless be necessary for special applications such as a substitute for text in a program aimed at the semiliterate. Some software producers have found ways to integrate tape recordings with computer software designed to drill spelling or a foreign language, or simply to introduce a neophyte to computer use. Such a circumvention of the speech synthesizer may be worth consideration if vocalization is a necessary part of the software being planned.

Interaction

By far, the most important and effective capability of the computer to motivate learning and to drill, remediate, teach, or model educational material is its capacity for immediate response to the user's input. Each student can be steadily and actively engaged in the practice of a skill, the development of an idea, the exploration of a concept, or the evolution of a problem-solving strategy. Students need not sit with hand waving in the air or wait days or even weeks between taking a test and receiving the results. It is unfortunate that more software does not take advantage of this computer capability. Programs that are presented like a book equipped with an automatic page-turning device or like film that rolls on regardless of viewers' attention should be avoided. In fact, it is useful when reviewing or creating software to be aware of the percentage of time the user will be actively engaged and to remember what mechanisms, if any, are built into the program to remind the student that some response is required. When developing a piece of software, developers should maximize user interaction within the confines of the educational objective.

Randomization

Many varieties of educational software call upon the computer's capability for randomizing words, numbers, questions, and so forth. Suppose a test poses ten questions but draws on a list of a hundred or more items. The items for the particular drill or test are selected by the computer's random number generator. This is a built-in function of the computer, but the accuracy of the randomization process varies from model to model. If the need for this capability is an essential feature of the software under consideration, it might be well to run some diagnostic tests on the random generator of the machine to be used.

Pedagogical Considerations

The entire program must be planned, or evaluated, to ensure that it provides an effective learning experience. This is the major and overriding concern. Even if the objective and task are well defined and creative use is made of the computer's special capabilities, it will mean nothing unless the program meets other standards as well.

Information Content

The educational content should represent the best knowledge currently available in the area and should be correct. As any experienced software reviewer knows, errors creep into programs as they do into textbooks. A recent run-through of a spelling game, for example, produced "concieve" as the correct spelling of "conceive." Two different programs designed to teach U.S. states and their capitals gave differing answers regarding the capital city of Kentucky. An otherwise humorless drill in vocabulary introduced each lesson as "EXERCIZE." More subtle perhaps is the following: An arithmetic drill posed the question "3/5 + 3/5." A student answer of "6/5" elicited a response of "INCORRECT." To a second try of "6/5," the computer responsed, "Please reduce." Clearly, the computer had been programmed to expect the answer "1 1/5," the correct answer in mixed number form. The student had, however, not been given the instruction to enter answers in mixed number form. More important, the answer "6/5" is not only correct but it cannot be reduced, although it may alternatively be expressed in mixed-number form.

Error Handling

Another pedagogical consideration, vital to the success or failure of a program, is, "How are incorrect student responses handled?" Some programs refuse to accept an answer in a different form and continue only to pose the question until the student has arrived at the programmed correct answer. Such treatment could be a model of frustration. At the other extreme, the program will respond simply "INCORRECT" and proceed to the next question or section of the exercise, leaving the student without any knowledge of the answer desired.

An effective option is to allow the error and illustrate the consequences. In a hypothetical paramedic training program, for example, the choice of "artifical respiration" rather than "tourniquet" for a slashed wrist would result in the death of the patient. The attractiveness of this approach is illustrated by the answer of one video arcade afficionado who was asked, "What is so great about video games?" He said, "You can die so many times."

Goals

While the actual goal of any instruction is student mastery of material, subsidiary goals can be introduced to make attainment of mastery more interesting. By way of example, the three programs described below each purport to increase speed and accuracy of typing.

The first is tutorial. The student is given information about where to locate the fingers of each hand on the keyboard and is then presented with sequenced drills. At the end of an exercise, a score is awarded reflecting words per minute and errors made. The student is then advised to repeat the lesson or to advance to the next drill. In this program, motivation to continue is sustained primarily by the student's desire for mastery.

The second typing program starts by asking the student to set his or her own goals both in words per minute (wpm) and permitted errors per line. A beginner may elect to try for 5 wpm with a generous allowance of errors. The more advanced typist may opt for 80 wpm with no errors. Progress through the sequence of drills is governed by whether or not the student-set goal is achieved.

Yet a third such program provides a sequence of drills, but the format is like an arcade game. Enemy craft approach the student's spaceship; to save the ship, the student must type letters, words, or phrases before the alien craft attacks.

The first of these typing programs is likely to attract only students who are highly self-motivated. For those who are afraid of failure and easily discouraged, the second version may be more appropriate. For others who might otherwise not have the self discipline to expend time and energy on drill, the arcade-like format might well be the most successful.

Many adult students and even educators support the notion that learning is "serious business" and that devices to make learning fun are, therefore, suspect. The attraction of games and the success of game-formatted learning experiences nevertheless have led software producers to market an increasing number of educational games. Research projects studying the educational value of electronic games tend to support the instructional value of such formats (Malone 1981). Although very few educational programs created specifically for vocational training are yet on the market, in considering purchase of those that are available or the development of new software for vocational purposes, the game treatment should be examined for its relationship to the intended educational objective as well as for its motivational value.

Is the gaming element merely a reward for successful performance? Does the correct answer, for example, simply provide the player with the opportunity to participate in the "game," permitting the shooting of a basket in basketball or the spinning of a spinner to determine a move in a board game? Or, is the format intrinsically related to the subject matter? Such a relationship exists in the arcade-type typing game where the goal of speed and accuracy is directly related to the pressure of "enemy" attack that increases as the player becomes more adept. Another example is an estimation drill based on bowling in which the better the player's estimate, the more pins are knocked down.

Both have benefits. In the former case, correct performance is encouraged because playing of the game is the more immediate goal. This approach, while denigrated by some, has practical value and can more easily be applied to a wide range of subject matter. Designing programs that incorporate intrinsic learning requires imagination, creativity, skill, and luck.

Time Use and Timing

How much student time must be spent in the completion of the projected computer program? How much time must be allowed for use of this program to be beneficial to the student? In judging or planning software, individuals should identify the specific learning objectives in this context. An automated accounting program designed as courseware for bookkeeping students may consume sixty hours of student time. Drill and practice can be effective with as little as ten minutes a day exposure to a computer program. Simulations can take varying amounts of time depending on the complexity of the problem, but most certainl' ccupy more than a single forty-five minute period. When computer use will, ı all probability, take more than one class period, instructors should ,lan for the recording of students' progress throughout the program so that they need not start from the beginning on succeeding sittings.

Timi ıg within a program must also be considered. Is there a specific limit on the amount of time allowed for student response? Is that limit reasonable or, better still, can it be preset by the student or teacher? What happens if the time limit is exceeded? If there is no limit, does the program have any signal (such as a beep or flashing screen) tu attract a user's attention to the fact that a response is required?

Language and Instruction

In software designed for school-aged children, the question of language can be dealt with roughly according to grade level of the students for whom the program is intended. This is not the case with adult learners. Very often even nonliterate adults will have a useful oral vocabulary of far greater sophistication than that of a third or fourth grade child. It is important to suit both language and content of software to an adult level of understanding and interest and not refer to "bunny rabbits" and "choo choo trains." On the other hand, it makes little sense to display instructions for the operation of the program that are outside or beyond the reading capability of the intended user. One solution is to teach operation of the program orally and ask users to pass along the instructions. When appropriate to content, this works well and can provide valuable enhancement of self-image to the user-teacher. Where instructions are necessary, text should be displayed with space between lines and without crowding the screen. Users must be allowed to read at their own pace. A mechanism, therefore, should be built in to allow them to advance to the next "page" by depressing a particular key.

Finally, safeguards should be built into the program so that, if instructions are not followed, the program responds appropriately. Incorrect execution of instructions should not "break" the program (return it to the beginning) but should result in further instruction to the user on how to proceed.

Test Use

Test use of a software package is essential. Not until material has been field-tested over a reasonable length of time, can one be certain that all the problems have been eliminated. Results of field tests should be made available to those who wish to purchase the software. If possible, purchasers should confer with an institution that is already using the software.

Software should only be purchased from a marketing source that allows a reasonable examination period. Fortunately, more and more software producers are offering to refund the purchase price if software is returned within a thirty-day period.

Societal Issues

Much criticism has been leveled in recent years at tests and workbooks because of inherent sexism, racism, or propaganda. Instructional material inherently reflects the prejudices of its authors. This is no less true of instructional software. The computer's interactive capability and its power to involve the user impose a responsibility on the purchaser or developer to be aware of any implicit teaching in a given piece of software.

One must at least ensure that computer response to user input is not derogatory or ego deflating. The nonjudgmental quality of the computer is much proclaimed as one of its virtues, but software is only as nonjudgmental as its authors.

Less apparent, but just as important, are undercurrents of propagandism. A game-formatted program that involves "blasting alien spacecraft out of the sky" could be conveying the message that other intelligent beings in our universe are to be regarded as hostile. In a commercially available and otherwise excellent software simulation of a presidential campaign, the user-candidate must take stands on issues, consult polls, allocate campaign funds, and otherwise engage in activities similar to those of a real campaign. Success in this simulation depends largely on dollars assigned to advertising and media exploitation. This says something about U.S. political races that may reflect the truth but may, nevertheless, not be desirable in an educational simulation.

Management Considerations

Instructors can reasonably expect a computer to aid in the management of instruction, that is, to keep individual student records, calculate individual and group averages, generate progress charts, and maintain and update other statistical information. Ideally, this kind of computerized record keeping would be integrated with a branching program that administers tests, assigns student work on the basis of the computer-evaluated test results, and conveys any and all results to the teacher on demand. This has, in fact, been achieved in the computer-managed instruction (CMI) packages marketed on mainframe and minicomputers by Control Data (its package is PLATO) and Computer Curriculum corporations, among others. To reproduce such complex integration of management and instructional functions, however, has not been satisfactory for the microcomputers.

Some management programs for microcomputers correlate student data, but teachers must regularly compile and feed in that data. Also, some sequenced curriculum programs test students and assign appropriate work, but currently little, if any, software for microcomputers successfully correlates these two functions.

The record-keeping capability of most educational software is limited to tracking of errors made during the use of the program and providing a summary on the screen at the end of the exercise or session. An option to print out this summary can be, and sometimes is, included. Such an option, which allows the teacher to let the student decide whether or not to report a score, has been found by some teachers to have a positive and motivating effect on adult student performance.

In evaluating the management capability of currently available microcomputer software, it should be kept in mind that, in the present state of the art, a barter situation exists: increased capacity for record keeping usually means decreased program flexibility.

Documentation

"Documentation" refers to the body of written material accompanying a software or a hardware package. Good software documentation should include the following:

- Clear marking on the outside of the package of the make and model of microcomputer for which the software is intended.
- Information on the copyright and licensing conditions of the individual package.
- A list, again on the outside of the package, of all equipment needed for the proper operation of the software including the memory requirements of the program.
- Step-by-step directions for activating the software. These should include instructions for the order in which each peripheral piece of hardware is to be turned on; and all other information necessary to ensure that an inexperienced purchaser can make the program run.
- Explanatory material relating to lesson content including a description of a sample run-through. For the teacher this material should include an outline detailing typical classroom and/or student assigned use, instructional level, teacher options, and management and editing capabilities. Material for the student should include a menu of options, the goal of the program, and what, if anything, the student must contribute to the interaction.
- If the material is to be edited by the teachers, instructions for editing designed for a person with little or no computer experience.

o Supplementary worksheets and textual material if dictated by content.

Summaries of previous evaluations of the software are valuable, as are reference sources. Occasionally, a particularly complex piece of software will include a "hotline" telephone number that teachers and/or students may call for information about operational problems they incur in using the software.

Manuals accompanying software and hardware have been severely criticized for their abstruseness, for the use of unexplained technical language, and even for blatant errors. Obviously, that documentation should be clear and accurate, but one should not assume that it will be.

A final and important note: Software (and hardware) should be fun to use. Through the use of computers, the learning process should not only be facilitated but also made more enjoyable. Before using a program in the classroom, educators should ask themselves, "Would I like to have this particular software assigned to me?"

CONCLUSION

Vocational and career training encompasses an almost limitless amount of subject matter, ranging from basic reading, writing, and computational proficiency to the mastery of highly specialized and sometimes quite complex technical skills. Microcomputers can be enormously valuable in helping meet the challenges of providing vocational and career education. At the same time, a dearth of suitable software meets the criteria set forth in the guidelines. Development of such software is clearly an urgent necessity, but until this happens, what is the educator to do?

Several approaches are possible. One is to use only general-purpose software. Offer training in word processing coupled with creative use of the word-processing software to develop reading and writing skills in a particular vocational area. Add instruction in the use of a spread-sheet program and create exercises that apply the capabilities of that program to a specific vocational skill. Include LOGO to introduce students to programming languages and enhance their problem-solving skills. Some students may become adept enough at programming to use LOGO to create simple and usable software routines for in-house drill and practice.

Another approach is to explore the body of public domain software that is available for most microcomputers.* While it is unlikely that much of this material will be suitable for vocational application without substantial revision, public domain software may be freely adapted and edited. The two "States & Capitals" drill programs referred to earlier are, in fact, commercial adaptations of a public domain program. By substituting different graphics, this same program could be made into a "parts of an engine" drill, for example. The editing process will be time-consuming and will require some programming expertise. The end product, however, can be effective in attaining a discrete learning goal.

Still another alternative is to invest in commercially available open-ended software, designed so that its content can be changed or supplemented by the teacher. One of the typing programs previously described is an example. The sequence of drills included in the program can be extended to provide typing practice with any specialized vocabulary list. It is only necessary that someone take the time to provide the words and, by following simple instructions, record them on the program disk. If useful formats can be

---

*Two sources of information about public domain software for a particular microcomputer are Ann Lathrop, SOFTSWAP, San Mateo County Office of Education, Redwood City, CA 94061; and Barbara Rozell, Resources in Computer Education (RICE), Northwest Regional Educational Laboratory, 300 S.W. Sixth Avenue, Portland, OR 97204.

found, a few such programs can be expanded into a considerable library of software.

A more adventurous path to the possible acquisition of software is suggested by a survey conducted by Kearsley, Hillelsohn, and Seidel (1981-82). By polling major corporations about their use of microcomputers for staff training and inservice education, they found that 50 percent of the respondents had developed microcomputer software for the following educational purposes:

- Technical Skills . . . . 71 percent
- Management/Duties . . . . 22 percent
- Administrative/Clerical . . . . 20 percent
- Programming/Computers . . . . 33 percent
- Management Training . . . . 21 percent

The companies responding to this survey represent varied fields. They include, among others, the Xerox Corporation, Boeing Aerospace, and AETNA Insurance. Clearly, considerable corporate-owned software exists that, although not yet available to the public, almost certainly could be useful to adults seeking vocational training. Perhaps pressure could be brought to bear on these companies to share their educational resources with adult education facilities in their communities.

Whatever short-term solutions to the software shortage are chosen, some computer use should be an important element in any vocational training. With the computer component of an adult education program consisting only of LOGO, a word-processing program, and a handful of short drill and practice routines adapted from public domain software, the learning experiences of students can be enriched and expanded. Even such a limited program will provide these adult students with contact and insight into the vital technology of the times.

REFERENCES

Ayers, G. "The Learner and the Computer." Paper presented at the National Conference on Computer Based Education, Bloomington, MN: October 23-25, 1980. (ERIC Document Reproduction Service No. ED 200 188).

Dennis, J. R. Evaluating Materials for Teaching with a Computer. The Illinois Series on Educational Application of Computers, no. 5e. Urbana: University of Illinois, 1979. (ERIC Document Reproduction Service No. ED 183 185).

Garrison, R. "Microcomputers and CAL in Adult Education." Lifelong Learning: The Adult Years 5 (June 1982): 22-23. (ERIC No. EJ 265 454).

Judd, D. H. "A Microcomputer Role in Adult Reading and Writing Skill Development." In Microcomputers for Adult Learning: Potentials and Perils, edited by D. G. Gueulette, pp. 70-85. Chicago, IL: Follett, 1982.

Kansky, B.; Heck, W.; and Johnson, J. "Getting Hard-Nosed about Software: Guidelines for Evaluating Computerized Instructional Materials." Mathematics Teacher 74 (November 1981): 600-604. (ERIC No. EJ 255 389).

Kearsley, G.; Hillelsohn, M.; and Seidel, R. J. "Microcomputer-based Training in Business and Industry: Present Status and Future Prospects." Journal of Educational Technology Systems 10 (1981-82): 101-108.

Little, D. Introduction to Teaching Adults. Module 5: Designing Instruction. Vancouver: University of British Columbia, Centre for Continuing Education, 1981. (ERIC Document Reproduction Service No. ED 208 205).

Malone, T. "What Makes Computer Games Fun?" BYTE 6 (December 1981): 258-277.

Martin, K. Microcomputer Assisted Instruction in Adult Basic Education. Lebanon: Indiana Vocational Technical College, [1982]. (ERIC Document Reproduction Service No. ED 222 682).

Meierhenry, W. C. "Microcomputers and Adult Learning." Training and Development Journal 36 (December 1982): 58-66. (ERIC No. EJ 270 787).

Olds, H., Jr. "Evaluating the Evaluation Schemes." In Evaluation of Educational Software: A Guide To the Guides, edited by N. Jones and L. Vaughn. Austin, TX: Southwest Educational Development Laboratory, and Chelmsford, MA: Northeast Regional Exchange, 1983.

Rahmlow, H. F. "Computer Facilitation of Training and Education." Paper delivered at the American Educational Research Association annual meeting 1979 as abstracted in "Surprise! CAI Really Can Be Cost-Effective." Training 16 (September 1979): 28.

Soloway, E.; Lockhead, J.; and Clement, J. "Does Computer Programming Enhance Problem-Solving Ability? Some Positive Evidence on Algebra Word Problems." In Computer Literacy: Issues and Directions for 1975, edited by R. Seidel, R. Anderson, and B. Hunter. New York: Academic Press, 1982.

Technical Education Research Center. Microcomputers in Education--Resource Lists. Cambridge, MA: TERC, 1982.

Ulmer, C., and Dorland, J. "Modern Adult Basic Education: An Overview." In Outlook for the 80's: Adult Literacy, edited by L. Mercier, pp. 1-22. Washington, DC: Dingle Associates, Inc., 1981. (ERIC Document Reproduction Service No. ED 211 701).

U.S. Department of Education. "Instructional Use of Computers in Public Schools." Washington, DC: National Center for Education Statistics, September 1982.

BIBLIOGRAPHY

Brockmann, F. "Identification and Evaluation of Educational Applications Software." In AEDS Proceedings: The Tomorrow in New Technology; Frontiers in Administrative Computing; Adventures in Instructional Computing. Washington, DC: Association for Educational Data Systems, 1982. (ERIC Document Reproduction Service No. ED 223 239).

Cain, D. J. "Computer-based Training at United Airlines." Training and Development Journal 35 (March 1981): 76-80.

Campbell, J. O. "2D Simulation: Educational Breakthrough." Interface Age 5 (October 1980).

Cohen, V. "Criteria for the Evaluation of Microcomputer Courseware." Educational Technology 23 (January 1983): 9-14. (ERIC No. EJ 279 832).

Cohen, V. "Evaluating Instructional Software for the Microcomputer." Paper presented at the annual meeting of the American Educational Research Association, New York, March 1982. (ERIC Document Reproduction Service No. ED 216 704).

Dearborn, D. E. "A Process for Selecting Computer Software." NASSP Bulletin 66 (September 1982): 26-30. (ERIC No. EJ 268 224).

Douglas, S., and Neights, G. Instructional Software Selection: A Guide to Instructional Microcomputer Software. Microcomputers in Education Series. Harrisburg: Pennsylvania State Department of Education, n.d. (ERIC Document Reproduction Service No. ED 205 201).

Dresden Associates. Evaluations of Educational Software for Apple, PET, TRS-80, with Index to Evaluations in Other Publications. Dresden, ME: Dresden Associates, 1981. (ERIC Document Reproduction Service No. ED 213 389).

Eisele, J. E. "Instructional Computing, Selecting Courseware." Educational Technology 23 (May 1983): 37-38.

Filliman, P. "Guidelines for Introducing Microcomputers in the Schools." Arithmetic Teacher 30 (February 1983): 16-17, 56. (ERIC No. EJ 275 044).

Gueulette, D. G., ed. Microcomputers for Adult Learning: Potentials and Perils. Chicago, IL: Follett, 1982.

Heck, W; Johnson, J; and Kansky, R. Guidelines for Evaluating Computerized Instructional Materials. Reston, VA: National Council of Teachers of Mathematics, 1981. (ERIC Document Reproduction Service No. ED 201 485).

Hilgenfeld, R. "'Checking Out' Software." Computing Teacher 9 (November 1981): 24-27.

International Council for Computers in Education. Evaluator's Guide for Microcomputer Based Instructional Packages, rev. ed. Eugene, OR: ICCE, 1983.

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

Jones, N. G., and Vaughan, L., eds. Evaluation of Educational Software: A Guide to the Guides. Austin, TX: Southwest Educational Development Laboratory, and Chelmsford, MA: Northeast Regional Exchange, 1983.

Kleiman, G.; Humphrey, M. M.; and Van Buskirk, R. "Evaluating Educational Software." Creative Computing 7 (October 1981): 84-86, 88-90. (ERIC No. EJ 252 712).

Lathrop, A. "Microcomputer Software for Instructional Use: Where Are the Critical Reviews?" Computing Teacher 9 (February 1982): 22-26.

Lewin, A. W. "Down with Green Lambs." Theory into Practice, forthcoming.

Lipson, J. I. "What Educational Computer Systems Should Be." Journal of Educational Technology Systems 8 (1979-80): 279-285.

Lipson, J. I. "Riding the Wave of the New Information Technologies: The Administrator's Role." EDUCOM Bulletin (Winter 1983).

Luehrmann, A. "Computer Literacy--What Should It Be?". Mathematics Teacher 74 (December 1981): 682-686. (ERIC No. EJ 257 003).

Luehrmann, A. "Computer Literacy." EDUCOM Bulletin (Winter 1981).

Mallory, W. J. "Simulation for Task Practice in Technical Training." Training and Development Journal 35 (September 1981): 13-14, 16-20. (ERIC No. EJ 250 780).

Naiman, A. "Computers in Education: The Question of Equity." School Microware Reviews 1 (Summer 1981).

New York Department of Education. "Considerations for Selecting Computer-based Instructional Systems." Albany: University of the State of New York, The State Education Department, 1983.

Northwest Regional Educational Laboratory. Evaluator's Guide for Microcomputer-Based Instructional Packages. Portland, OR: NWREL, 1981. (ERIC Document Reproduction Service No. ED 206 330).

Perry, T., and Zawolkow, G. "CAI: Choosing Hardware and Software." Apple Orchard 3 (March 1982): 22-24.

Price, R. V. "Selecting Free and Inexpensive Computer Software." Educational Computer Magazine 2 (May-June 1982): 24-26. (ERIC No. EJ 268 631).

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.

Southwest Educational Development Laboratory. R&D Speaks: Evaluation of Educational Software. Austin, TX: SWERL, 1983.

Tinker, R. "Microcomputer-Based Experiments." Paper distributed at National Science Foundation Committee Meeting, April 1983.

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