DOCUMENT RESUME

ED 217 872 IR 010 258

AUTHOR Becker, Henry Jay
TITLE Microcomputers in the Classroom--Dreams and Realities. Report No. 319.
INSTITUTION Johns Hopkins Univ., Baltimore, Md. Center for Social Organization of Schools.
SPONS AGENCY National Inst. of Education (ED), Washington, DC.
PUB DATE Jan 82
GRANT NIE-G-80-0113
NOTE 80p.

EDRS PRICE MF01/PC04 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; Computer Managed Instruction; Curriculum Development; Drills (Practice); Educational Environment; *Elementary Schools; Elementary Secondary Education; *Instructional Systems; *Microcomputers; Pilot Projects; Problem Solving; Program Effectiveness; Programing; *Secondary Schools; Simulation; Tutoring
IDENTIFIERS BASIC Programing Language

ABSTRACT

Taking the current excitement among educators concerning the uses of microcomputers for student instruction as a point-of-departure, this paper addresses the problems and possibilities associated with the uses of microcomputers in the classroom and discusses these in terms of instructional techniques and the social aspects of integrating computer activities into classroom environments. A pilot program in which microcomputers were used to teach programming in elementary school classes is described and the six major instruction-related uses of computers--drill-and-practice, tutorial computer-assisted instruction, computer-managed instruction, simulation and model building, the development of computerized information skills, and teaching computer programming--are carefully reviewed. Special attention is given to the place of computer programming in the school curriculum and the advantages and disadvantages of the BASIC programming language are outlined. The social organization of computer use in schools is then considered. A summary of recommendations for researchers, developers of computer-based educational materials, and school system administrators concludes the paper. A 45-item reference list is attached. (Author/JL)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED217872

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it
Minor changes have been made to improve reproduction quality

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy

Microcomputers in the Classroom--Dreams and Realities

Grant No. NIE-G-80-0113

Henry Jay Becker

Report No. 319

January 1982

Published by the Center for Social Organization of Schools, supported in part as a research and development center by funds from the United States National Institute of Education, Department of Education. The opinions expressed in this publication do not necessarily reflect the position or policy of the National Institute of Education, and no official endorsement by the Institute should be inferred.

Center for Social Organization of Schools
The Johns Hopkins University
3505 North Charles Street
Baltimore, MD 21218

Printed and assembled by the Centers for the Handicapped
Silver Spring, MD

IR010258

Introductory Statement

The Center for Social Organization of Schools has two primary objectives: to develop a scientific knowledge of how schools affect their students, and to use this knowledge to develop better school practices and organization.

The Center works through five programs to achieve its objectives. The Studies in School Desegregation program applies the basic theories of social organization of schools to study the internal conditions of desegregated schools, the feasibility of alternative desegregation policies, and the interrelations of school desegregation with other equity issues such as housing and job desegregation. The School Organization program is currently concerned with authority-control structures, task structures, reward systems, and peer group processes in schools. It has produced a large-scale study of the effects of open schools, has developed Student Team Learning instructional processes for teaching various subjects in elementary and secondary schools, and has produced a computerized system for school-wide attendance monitoring. The School Process and Career Development program is studying transitions from high school to post-secondary institutions and the role of schooling in the development of career plans and the actualization of labor market outcomes. The Studies in Delinquency and School Environments program is examining the interaction of school environments, school experiences, and individual characteristics in relation to in-school and later-life delinquency.

The Center also supports a Fellowships in Education Research program that provides opportunities for talented young researchers to conduct and publish significant research, and to encourage the participation of women and minorities in research on education.

This report, prepared by the School Organization program, examines the current state of microcomputer use in schools and classrooms, the potential for further use, and the problems involved in achieving the potential.

Abstract

This paper addresses problems and possibilities associated with uses of microcomputers in the classroom and discusses these in terms of instructional techniques and social aspects of integrating computer activities into ongoing classroom environments.

Widespread excitement about using relative inexpensive desktop microcomputers for student instruction needs to be restrained both by consiòerating how computers might ideally help young minds develop and by considering how actual implementations of microcomputer equipment in classrooms are likely to be made in the near future. The paper concludes with a summary of recommendations for researchers, developers of computer-based educational materials, and school system administrators.

ERIC

> "We should not predict or expect that the personal computer will foster a new revolution in education just because it could. Every new communication medium of this century--the telephone, the motion picture, radio and television--has elicited similar predictions that did not come to pass. Millions of uneducated people in the world have ready access to the accumulated culture of the centuries in public libraries, but they do not avail themselves of it. Once an individual or a society decides that education is essential, however, the book, and now the personal computer, can be among the society's main vehicles for the transmission of knowledge." (Kay, 1977)

In the last two years, many educators have become excited about using the new generation of small, desktop computers, called "microcomputers," to assist teachers in classroom instruction and to broaden students' intellectual experiences. This excitement has been felt even by teachers and administrators who have never touched a computer. But it has also been shared by curriculum developers, computer scientists, and others who have tried for many years to apply the capabilities of digital computers to educational purposes and settings. The new microcomputers, although no more computationally powerful than machines on which computer-assisted instruction was first developed nearly two decades ago, are simpler to use, allow more responsive and interactive student-and-computer dialog, and, most significantly, are far lower in cost.

Despite this excitement about using the new microcomputers in schools--an excitement that is reinforced by commercial efforts to sell computers and computer program products to this potentially large and relatively centralized market--many people doubt that single or multiple purchases of instructional equipment costing hundreds or thousands of dollars per item can be justified in most schools unless unprecedented academic accomplishments can be expected

ERIC

to follow from their use.

There is reason to be skeptical, for example, that the current Apple and Radio Shack computers which many schools are purchasing are cost-effective devices for providing the remedial drill and practice of basic math and language skills for which the commercial advertisements so willingly tout them. Although computers do provide opportunities for individualization, immediate feedback, and summarization of individual student performance that other methods of skills practice may lack, it is not clear that these computers enable skills to be learned so much more rapidly that their investment is worthwhile on these grounds alone. The potential capacity of computers to diagnose student error patterns and provide corrective tutorial instruction--features which might more reasonably make computers cost-effective teaching aids--have rarely been demonstrated in current microcomputer-based learning materials.

However, it is not only the content of programs available for currently marketed microcomputers that raises questions in many minds. There are also important organizational and curricular problems to solve before the technology may be effectively used to increase learning efficiency in math and other subjects.

The most obvious organizational problem is that while most products developed for classroom uses of microcomputers allow profitable use by only one child at a time, schools typically purchase microcomputers in very small quantities--most often a single computer in a single classroom, perhaps later expanded to three or four. For teachers to effectively use a device which is primarily geared for a single user when they have one machine and 30 children demands, at a minimum, a major planning effort.

ERIC

Besides these practical reasons for skepticism about using computers in classrooms today, there are both empirical and ethical questions about the longer-term prospects for computer-use in classrooms. If "computer-assisted instruction" is still to be the primary function of computers in the classroom, one must ask whether providing a new method for having students practice applications of rote-learned rules of grammar and arithmetic is more important than using school resources to develop more higher-level intellectual skills of students.

Also, some people have questioned whether increased computer-based instruction will replace rather than supplement the learning time that is spent in an interpersonal context. If so, it may result in higher academic performance at the cost of diminished social skills. In large segments of the public, anything with the name "computer" is denigrated as inevitably producing mechanical, impersonal schooling; thus, the manner in which schools publicize their computer-related activities may also be a problem.

This paper explores both the DREAMS that are held for computers in the classroom and some of the REALITIES that these dreams must confront. In a future paper, I will suggest some positive and concrete considerations to guide the behavior during the next several years of those who would build computer-based learning tools and those who would purchase them. Throughout both papers, I try to keep in mind two basic questions: "What uses, if any, should we make of today's microcomputer products in today's classrooms?" and "What can we do to make microcomputers more useful to schools during the next five years?"

This practical focus is important because every week schools and school systems across the country and manufacturers and developers of educational materials are making decisions which will affect the ability of computers to assist (or impede) the education of thousands of students. On the other hand, focusing only on the immediately possible is unnecessarily limiting. Without expanding one's attention to some of the more imaginative images of how computers might be able to function in education, there will be a tendency to evaluate current attempts to use computers in too limited a context. So I will begin with some dreams--some of the ways that computers might be used by children and adolescents in school settings perhaps ten to fifteen years from now.

Dreams

Imagine a dozen second-graders seated in their classroom, each conversing with a personal "videopal." They speak to it in lowered voices, as they have been taught by their teacher, and their videopal, in turn, speaks back to them--instructing them, for example, to read aloud the next word or phrase displayed on the screen. The computer, personalized as a videopal, meanwhile assesses the child's performance level, recalls from its stored data information about the child's recent particular problems, and selects an appropriate stimulus for the next reading task. After the response, the machine examines the response for errors, announces gently how the words were to have been pronounced, and provides additional opportunities or examples as follow-ups. Meanwhile, the classroom teacher is circulating through the room, assisting students with problems that the computer lacks the flexibility to handle and helping the students to use this electronic learning tool effectively. Elsewhere in the school, fourth and fifth grade students are engaged in similar "dialogs" with "intelligent computer tutors," responding by using the keyboard that they had learned to master during the previous year.

The computers in this example communicate by using a stored body of knowledge of both subject-matter and teaching method, emulating the behavior of a master teacher--knowing what kind of stimulus to present, how to evaluate the response, and what kind of feedback (both substantive and affective) to provide to assist the child's understanding and to maintain motivation for receiving future stimuli.

Now imagine a child in the same second-grade classroom--one who has become interested in the musical patterns and colorful designs that have been programmed into a "videobrush" computer available in the classroom learning center. Through the child's intrinsic interest in musical sound and color, she has learned to manipulate this device (which was programmed, of course, to encourage this manipulation) to create a music-and-color design of her own construction.

At the same time that the computer allows this "play" (i.e., self-directed activity not designed to yield a particularly known and generalizably useful skill), it explains to its user certain properties of color and sound and the human physiological apparatus that senses them, and it provides modifications of the child's construction and queries her about how these modifications might affect the result. A brilliant intermixture of "instruction" and "play" maintains the user's interest, develops creative thinking, and assists in an understanding of a subject-matter beyond the usual range of the second-grade curriculum and even beyond the capacity of the teacher to explain or perhaps even to understand.

There are many other such dreams. For example, one image is of a multi-faceted computer display screen which the student (probably of high school age) has learned to manipulate to be a writing pad, a calculator, a dictionary, a thesaurus, and "windows" on a variety of reference works and news and information sources. The student develops expository skills and information retrieval skills using the kind of "paper and pencil" which will be available to her in the real world after schooling. She manipulates the keyboard in front of her with an agility coming from several years of experience using keyboard communications devices. She uses special keys to get the computer's help when she cannot remember or does not know how to use the computer facilities to accomplish a desired retrieval or information manipulation. The teacher, meanwhile, is freed to provide more scholarly assistance, helping the students improve their expository style and contributing editorial judgment, leaving the computer to assist the student with some of the more mechanical aspects of writing, such as syntax and spelling.

In another image, a high school physics teacher is trying to develop in his students an understanding of the history of scientific understanding of physical principles. Using a large-screen video display which the entire class can see, he demonstrates the behavior of objects sliding down an inclined plane with the objects' behavior conforming to an assumed mathematical equation. The objects, of course, are not real, but dynamic visual images which look real to any but the most discriminating observer. After learning how a mathematical equation relates to the object's behavior, the students are asked to suggest variations in the equation, and they come to a judgment about what equation best conforms to the behavior of objects in the real world. The teacher mentions in passing that without the computer's ability to create these simulated worlds where things behave differently than in the real world, the students could only accept the equations on faith, because proving them would require experiments much too complex and expensive to be done in a classroom.

In a final image, we find an average high school student of the future who has become bored with using the computer in pre-programmed ways to practice reading and language skills, create musical patterns, write school reports, calculate solutions to math problems and examine alternative simulation models. Instead, having learned the immense variety of information manipulation of which the computer is capable, the student is busily engaged in creating new functions for the computer for use in his own life or for someone else's.

This capacity to combine known computer capabilities into a new function or product is what is generally known as programming a computer. Many people believe that such skills are beyond the capability of the average adolescent, but those who have worked with students--as young as age 5 or 6--have come to the opposite conclusion. Programming, although requiring capacities of logical thought and abstract reasoning, itself generates intellectual growth in these skills, and is simply not quite so foreign to those who have used the computer in a variety of ways as to those not experienced in doing so.

Programming both requires and generates the capacity to perform logical operations on pieces of information, to structure information in a systematic way, and to creatively combine different pieces of information in a new and functional way. Low-cost computers, by being more available and accessible, may enable these capacities to be generated in a much broader population of adolescents than previously.

Each of these dreams is impractical today. Each demands from available computer technology, from commercial or internally-developed computer programs, or from teacher-possessed computer knowledge more than almost any school is able to obtain or provide. Nevertheless, the images contain most of the goals that those excited about the possibilities of computers in the classroom have in their minds:

(1) The ability of intelligent and communicative machines to provide appropriate instructional stimuli on an individualized basis, providing diagnoses and feedback both to the student and to the teacher-manager responsible for monitoring the student's progress.

(2) The creation of intellectually stimulating environments for teaching subject-matter generally foreign to the current curriculum, perhaps beyond the competency of the teacher, but important and useful preparation for the world of the future.

(3) The resources of a highly complex but flexible information storage, retrieval, and processing machine--comparable to a library, a librarian, a typewriter and an editor--all accessible to the student and together providing necessary skills for the student's subsequent adult experiences.

(4) An appliance able to provide experiences and opportunities to students through simulations which would be otherwise too costly, too risky, too time consuming or not possible. People often learn best by participating in a system rather than merely being a spectator. Computers provide a way to get close to actual participation without the costs or risks of actual participation.

(5) A generation of young adults performing logical thought, processing information and performing analytic tasks far better than previous ones--due to an early and continuous exposure to concepts and methods of computer programming.

Although these classroom images are highly idealized, it is possible now to develop materials so computers could function in ways that resemble these scenes. Programs can be written and used to have children and adolescents explore music and art theory, to be tutored in arithmetical operations, to practice retrieving information from a computer's "data-base," to simulate social environments and scientific principles, and to test students' models of these systems; and classes can be organized to teach students to program computers.

Yet practical implementations that use such programs, limited by available technology and funds, by the methods used to implement them, or by external constraints such as teacher skills or institutional inertia, could all be pale imitations of the ideal and may deliver none of the promises so appealing in the original image. It is important to consider which kind of implementations schools are able to make today using available equipment, realistic budgets, and current manpower and expertise, and to begin with these limitations to develop materials and methods of implementing them that will optimize the contributions that current microcomputers can make to life in schools.

The Realities

About 45 million students are enrolled in elementary and secondary schools in the United States today. These are grouped into about 2 million classrooms and one hundred thousand schools. Somewhere between 15,000 and 30,000 schools have at least one microcomputer or terminal connected to a larger computer--about one out of every four schools. The total number of such computers and terminals can be reasonably estimated to be between 20,000 and 60,000--about one computer for every 33 to 100 classrooms.[1]

[1] These estimates are based on a variety of sources including a survey by the National Center for Education Statistics, surveys by two firms who sell specialized lists of school personnel for marketing products, and discussions by persons in the field. The Center for Social Organization of Schools will be conducting a national survey of schools during 1982 and will be able to provide much more detailed accounting of possession and use of microcomputers in public and private schools by the Fall of 1982.

Of these classrooms with computer facilities, more now have stand-alone microcomputers than terminals connected by telephones to larger computers elsewhere (National Center for Education Statistics, 1981). Most of the terminal systems are older and many are being replaced by microcomputers. (Infoworld, DEC 7, 1981. p. 9.)

The number of schools with microcomputers is increasing at the rate of about 30% to 50% per year, but the numbers are still small compared to the number of students or even the number of classrooms. Even with a 50% annual increase in microcomputers (and no change in the number of functioning older pieces of equipment), by 1986 there will be only about three computers per school or one for every eight classrooms. It is important to keep figures like these in mind--even if they are only vague forecasts--when considering how computers can or will impact on classroom education.

For example, even if schools or districts commit one percent of their operating budgets for five years in a row to purchase microcomputer equipment--this is about half of what most schools spend annually on books, workbooks, and the like--by 1986 there would still be only one or two computers per classroom. Even at that rate of spending, which is beyond what is expected by the computer industry, there would hardly be enough computer equipment to drastically alter the way most instruction is given to the average student.

Anecdotal published reports of teachers using computers in their classroom, as well as published statistics on computer ownership by schools, confirm that schools typically acquire computer equipment for instructional purposes when a single teacher undertakes responsibility for management of a single microcomputer. There are some instances of administrative initiative, and certain states or school districts are pilot testing the use of microcomputers, but the most common etiology is for an enthusiastic teacher to obtain local school funds to purchase a microcomputer for his or her classroom. Secondary mathematics teachers appear to be the most frequent enthusiasts.[2]

A PILOT INTRODUCTION TO CLASSROOM COMPUTING WITH ONE MICROCOMPUTER

To see how typical teachers would use a single microcomputer in the context of classroom-organized learning, during the 1980-81 school year the School Organization Program at The Johns Hopkins Center for Social Organization of Schools provided six local teachers with a Radio Shack TRS-80 personal computer for a several week period. The computer was used in three upper elementary classrooms and three junior high mathematics classrooms. It was used primarily to expose students briefly to computer programming activities, rather than as a tool for students to practice their regularly taught math skills. This pilot effort was evaluated informally through periodic observation and conversations with teachers during the time they used the computer.

The students ranged from somewhat-below-average in current math performance to somewhat-above-average. In one of the six classrooms, the same students used the classroom for most of the day. In the rest, students had access to the computer only when their class was scheduled, generally for about an hour per day. The computer was present in each classroom for a three week period.

---

[2]For example, Brudner (1981) took a census of computer purchases in four states. In a more or less typical state, Oklahoma, he found that in late 1980, 40% of Oklahoma high schools had at least one computer. But few elementary schools had any. Most schools had exactly one computer. Very few had more than five. Most were in use in high school math and business education classes.

None of the participating teachers had had previous exposure or experience with computers. Three of the teachers were given a three-day workshop in Basic Programming prior to the fall semester. The other three participated in a 16-hour workshop in programming provided by the school district. All teachers had an opportunity to have the computer at their homes for a full week before they used the computer in their classroom. The teachers appeared to have a modest but sufficient understanding of programming techniques and vocabulary by the time they used the microcomputer in the classroom.

In each case, the teachers decided how they wanted to present lessons using the computer. They developed dittoed classroom assignments, homework assignments, and tests modeled on some of the materials that previous teachers borrowing the equipment had produced. The teachers were provided with a few programs for computer-assisted mathematics instruction, but none chose to use them because they did not provide practice on the specific skills currently being taught.

The students were taught programming using the computer language "BASIC." Previous experience with younger students had suggested that 5th grade was approximately the youngest grade level at which BASIC programming could be taught with the usual methods and expectations. This was confirmed in this pilot project--the 7th grade students made significantly greater progress in the three weeks of classroom use than the elementary school students. On the other hand, the class of 9th grade math students showed a much lower level of engagement in computer interaction and did not progress through the material as well as the seventh-graders. Overall, student progress in learning computer programming over the three week experience was limited, at best.

ERIC

Most instruction was provided by teacher lecture followed by individual seat-work and by individual or group work at the computer keyboard. Because of the student-computer ratio (approximately 30-1), each student had about 10 minutes on the average of actual computer time over the three weeks of "access" to the computer. Clearly, the learning was primarily through seat-work and other traditional means rather than through interactive work at the computer. The computer served mainly to verify that one's product (a computer program) could actually produce an outcome (computer "behavior") that could be observed. The variety of actual experience "on" the computer was drastically limited by resource availability.

The two teachers who taught groups of children who varied widely in current mathematics performance both found clear differences in how quickly the groups could learn the programming material. Both had been asked to use their judgment in providing the programming unit to the lower-performing student classes, but both chose to do so apparently for two reasons: they regarded the unit as an enrichment opportunity that they did not want to deny the slower-learning groups; and having to learn a new subject-matter themselves--computer programming--they felt it was too time-consuming to have to continue to prepare regular lessons for the slower groups at the same time they were developing the programming lessons.

Summarizing their experiences, all teachers expressed satisfaction that they had interrupted their regular curriculum to provide the computer's "enrichment." Most found the intellectual demands on themselves to be great, but worthwhile. Every teacher expressed the opinion that a single computer was insufficient to allow students to progress at the pace they could assimilate the new material. This insufficiency of resources for class-based computer instruction was the most important and most universally felt problem.

Despite the difficulties they had experienced with trying to integrate the use of a single computer with classroom-level instruction, several teachers subsequently tried to obtain a computer for their school. In each case, only a single computer was contemplated, due to anticipated financial constraints.

The problems faced by these pilot-test teachers may not be representative of the problems that most teachers face when trying to use a computer in their classroom. But certainly student access time must be a significant problem that arises in all but the most elaborate implementations of computers in the classroom--regardless of whether the computer is being used to teach programming or to provide drill work. In fact, using the computer to deliver instruction or practice in basic skills typically requires a higher ratio of computer stations to students than does using the computer as an "object of instruction." This is because programming activities involve a lot of "seat-work" such as learning appropriate vocabulary, planning a programming solution to a problem, writing the programming code, and "debugging"--finding errors that show up when the program is actually run at the computer.

SIX MAJOR INSTRUCTION-RELATED USES OF COMPUTERS

Having too few computers to effectively organize classroom instruction around them is not the only problem faced by schools trying to implement an instructional program that involves the computer. Other important issues faced by these innovating schools are (1) the marketing sophistication of educational merchandisers in contrast to a comparatively inexperienced user population; (2) the limited capabilities of the programs sold as computer-assisted-instruction and the limited capabilities of the computers bought by the schools to use for instruction; and (3) the almost complete lack of knowledge about computers by practicing teachers and the likely existence of this deficiency for perhaps another generation.

Some of these problems are general, affecting schools and teachers regardless of how they attempt to use computers in their classroom. But many of the problems are related specifically to the particular way that teachers attempt to use this versatile tool. Most instructional uses of computers can be considered under six headings:

(1) DRILL-AND-PRACTICE: Using computers for student practice of skills whose principles are taught by the teacher in traditional ways;

(2) TUTORIAL DIALOG: Using computers to present information to students, diagnose student misunderstandings, and provide remedial instructive communication, and individually-designed practice;

(3) MANAGEMENT OF INSTRUCTION: (tied either to computer-based drill-and-practice or to a separate test-scoring system--or independent of either one.) Using computers to provide the teacher with automatic reporting of individual student performances and appropriate assignment of skill levels;

(4) SIMULATION AND MODEL-BUILDING: Using computer programs to demonstrate the consequences of a system of assumptions, or the consequences of varying an assumption, usually in conjunction with instruction in science or social studies;

(5) TEACHING COMPUTER-RELATED INFORMATION SKILLS: Using the computer to teach students and have them apply such skills as typing, editing text, and retrieving information from computer systems;

(6) TEACHING COMPUTER PROGRAMMING: Having students learn to program computers to solve problems that are a part of their mathematics curriculum or simply for the understanding of programming itself;

This list of educational uses of the computer could be stretched, but these six encompass the most common current expectations that people have for classroom instructional applications.

COMPUTER-BASED DRILL-AND-PRACTICE

Both computerized drills and tutorials are variants of a general category of computer use called "computer-assisted-instruction," or C.A.I. The element common to all C.A.I. is repetition of similar exercises and immediate reinforcement (cognitive and affective feedback regarding performance). The more elaborate drill programs divide the instructional program into discrete tasks where progress from one task to another depends on a student's prior performance. Tutorial programs also include (1) an analysis of incorrect performance, (2) detailed instructional feedback to the student and teacher, and sometimes (3) the use of a drill designed to overcome the identified weakness. Tutorial programs typically minimize repetition of exercises in favor of a more rapid sequence of problem, evaluation, feedback, and sequence determination.

For most people, using computers for instruction in the classroom is synonomous with computer-assisted-instruction. Programs to drill students or to examine student responses and provide appropriate feedback are the most easily conceived applications of computers for classroom use. Most research on computers in education has centered around C.A.I. curricula and delivery systems; and most products marketed for educator-consumers have been C.A.I. products.

There are several reasons for this "natural" association between computers and traditional instructional activities for teaching math, reading, and language. First, these basic skill subjects require repetitious practice and memorization of isolated facts and skills. The most limited resource in

ERIC

the classroom--the teacher's attention and response to individual student needs--is often taken up directing and monitoring this practice activity, instead of pursuing other instructional efforts that require the broader and more flexible intelligence that people--not computers--can best provide. Having a computer present problems, accept student responses, provide feedback, and measure student attainment is one way to free the teacher for these more uniquely "human" functions of instruction.

Of course there are other procedures for drill that do not require constant teacher attention. Individual student workbooks and ditto sheets allow simultaneous student drill at a cost which may be a heavy burden on discretionary school budgets but which is substantially less than the cost of computer-assisted-instruction. Programmed versions of these workbooks--providing feedback to the student and routing the student to remedial instruction or the next instructional level depending on performance--would appear to be superior to regular workbooks, but they require more cooperation from students and are more expensive to produce. Programmed instruction was popular some years ago, but enthusiasm waned when research failed to show substantial student improvement over other teaching methods. In addition, heavy reliance on programmed instruction was boring to students--a common problem of most individualized instruction (Morgan, 1978).

Computer-based programmed drills hold several advantages over less automated forms. Their interactive nature and their flexible and visually appealing display formats make them more enjoyable experiences. Student feedback is direct and immediate. Automated management can be built in to give the teacher a summary of the student's achievement. And whereas traditional programmed instruction usually directs students to subsequent stimuli based on the correctness of the immediately preceding response, the computer can use a summary of the student's entire history of problem-solving to select the best route for that student. Of course, not every drill and practice program builds in all these capabilities.

Another important reason for the strong "historical" tie between drill practice and "computers in the classroom" is that compared to other types of programs, drills are easy programs to write--particularly in mathematics or in subjects where multiple choice items can be used. The production can even be automated, so that teachers can "write" drill programs without learning to "program" the computer in a traditional "procedural" computer programming language. Of course, the more automated production systems and programs dashed off with little effort are less likely to include precisely those features which give computerized drills an instructional advantage over alternative methods--management and teacher feedback and variability in display format and computer-to-student communication.

Because drill-and-practice programs are easier to write than most other kinds of instructional programs, and because many standard areas of the school curriculum involve a lot of repetitive practice of examples of discrete concepts, both supply and demand make drill-and-practice the most commonly produced computer-based learning materials. These two factors along with the greater standardization of the curriculum in the lower grades, have channelled most educational software toward the younger grade levels. This is so despite the fact that market surveys show clearly the most frequent purchasers of microcomputers for the classroom are <u>high schools</u>. Apparently, producers of software (computer programs) are trying to create a market for C.A.I. in the lower grades rather than trying to develop products for small, but existing, markets in the upper grades.

Research on Effectiveness

Drill-and-practice may also be attractive because major, systematic, well-monitored evaluations have found that computer-based drills can produce increased student achievement in comparison to prior years or in comparison to control group treatments. These positive research results have been found often in mathematics instruction, less frequently in language learning.

Vinsonhaler and Bass (1972), reviewing computer-based drills that pre-dated microcomputers in the classroom, concluded that C.A.I. drill-and-practice at the elementary school was more effective than traditional instruction in raising standardized test scores in the majority of 30 experimental comparisons at 10 sites.

Jamieson, et. al., (1974) found that C.A.I. drill programs are particularly effective for students who start below grade level. They conclude that at the elementary level, C.A.I. is "apparently effective as a supplement" to regular instruction, although they imply that it may not be more cost-efficient.

ETS's evaluation of the drill-and-practice curriculum tested in the Los Angeles schools (Ragosta, et. al., 1981) found that C.A.I. had a positive impact on computational mathematics, but increases in conceptual understanding of math were sometimes greater and sometimes less than for control group students. Also, language and reading results were less positive than the mathematics results, although students in the middle elementary grades did show strong improvement. The overall superiority of the results for math compared to language arts may be due either to differences in the ability of computer-based drills to provide useful practice or to differences in the quality of the particular C.A.I. curricula used.

However, this difference parallels the results generally found in the literature. Melmed points out that C.A.I. more closely simulates practice

ERIC

of math than practice of reading or language use. Equally important, perhaps, is that language and reading are used throughout the school day; thus C.A.I. language exposure is more limited relative to total exposure than is C.A.I. math experience (Melmed, 1980).

Even if computerized drills are more effective than traditional direct instruction, the issue of cost must still be addressed. At some point, the decision must be made about how much money more efficient progress in math or language achievement is worth. Clearly, for example, individual human tutors would produce increments in student achievement higher than that obtained by traditional classroom teaching. Just as clearly, school systems do not have the money to provide a tutor for every child. Computer use must be put to the same test. The computer must show a magnitude in improved learning efficiency that justifies the money investment and the investment of time and effort.

If we defer for the moment the concern about cost-effectiveness and limit ourselves to the problem of educational effectiveness, we still must question whether computerized drill programs will be effective additions to classroom instruction. The limited evaluation research shows that computer-based drill programs can be effective--given enough equipment for each child to have sufficient access and given appropriate content, organization of classroom activity, and monitoring. However, most of this research has been done under organizational conditions that allowed many computers to be in use at one time. Most involved use of time-sharing computer terminals rather than independent microcomputers, and were heavily monitored and well-managed implementations. Research should be conducted to determine whether most of the more typical drill-and-practice materials available for the TRS-80's, Apples, and other microcomputers the schools are now buying are

ERIC

as educationally effective under more typical conditions of use as were the pioneer C.A.I. programs.

Problems With Microcomputer Software

Two defects appear to predominate in current microcomputer software. First, most educational software is written in short, disconnected modules that are unrelated to one another and not clearly tied to other instructional activities or to specific textbooks. Second, most programs are boring and repetitious.

The first defect is due primarily to how the software is produced. Producers of most software for schools fall into four groups. First, teachers develop programs for their own use and then sometimes try to market their product. Much of the program library of the Minnesota Educational Computing Consortium has been built up in this way. Secondly, some program products are developed by programming teams within central offices of large school systems. The work of the Dallas and Philadelphia school systems, for example, is marketed by Radio Shack for their microcomputers. These efforts tend to be on a grander scale, but are less frequent.

Third, microcomputer drill-and-practice software for schools is produced by a large number of non-teaching programmers who use their home computer as a source of additional income. Sometimes these programmers combine into teams and take on a more professional public image, but most product lines are the work of one or two individuals. Fourthly, corporate enterprises, primarily publishers, sometimes form software development teams of programmers and curriculum designers in-house, but more frequently they buy out or distribute the products of independent program authors. The publishers' drill-and-practice products are a mixture of small stand-alone programs and programs that attempt to provide a complete package of instruction in a single subject for a wide range of initial achievement levels. Still, even these full-scale packages tend to provide limited approaches to the subject-matter which they do cover. The programming effort required is still large, even for drill-and-practice, and the pressure to market a product covering as much curricular territory as possible limits efforts to produce variety in the topic areas that are covered.

Two other related factors--low risk and low budgets--may contribute to the production of curricular fragments rather than complete, text-integrated tools. Due to major concerns about software copying, most publishers and distributers do not provide examination copies of educational computer programs. Thus, schools must often make decisions based on written advertisements and program descriptions. It is less risky to purchase individual programs costing, say, under $40 than to purchase a complete package at $400. Also, local budgets cannot usually absorb major software purchases, particularly after so much money was expended on the computer hardware! If schools emphasize hardware purchases more than software purchases, they may end up having computer equipment with capabilities that they cannot afford to implement. The survey of school microcomputer users planned by our Center during 1982 will provide more data about the etiology of school decision-making in this area.

The second practical deficit of most published computer drill materials is that, like the programmed instructional workbooks they often emulate, they tend to be boring and repetitious. The ETS's recent study of C.A.I. in Los Angeles schools found "boredom" as the objection to the program most often voiced by the participating children (Educational Testing Service, 1979). Observing children engaged in computer-based drills, one gets the clear

ERIC

impression that for them it is merely another assignment performed with little enthusiasm or care. In contrast, the same children are excited and enthused using the same computer to solve a puzzle or avoid an enemy's gunfire. Some drill implementations do attempt to combine the learning exercise with play activity. Banet (1978) reports the following two examples of game-environment-enhanced drills:

(1) An "estimation game" where feedback about accuracy of estimates is given by showing the distance from the bullseye of a target. Points are scored based on accuracy and speed of the estimate. This activity can be played by multiple players in competition with each other, or solitarily, in competition with previous high scores.

(2) Files containing the recognized vocabulary of individual students are used as individually-tailored data bases for word games. Each file can be automatically augmented by words used in the students' own computer-entered textual compositions.

The question is whether it is possible to create effective drill activities which minimize boredom and are as engaging as the play activities that the children enjoy. Based on an extensive analysis of computer games that children find "intrinsically motivating," Malone (1980) anticipates creating environments that maximize both learning and enjoyment.

Malone suggests that three factors be considered when designing motivational learning environments: first, the activities should be built around fantasy images that are meaningful to the intended audience. If possible, the learning required of the student should be "intrinsic" to the fantasy rather than simply added on. The example which Malone used in his research was a modification of a numerical estimation game developed originally for the Plato time-shared classroom computer where students "throw" darts to pop balloons along a number-line by estimating the point along the line at which the balloon was placed. In addition, Malone suggests that because different fantasies appeal to different people, the learning activities should be imbedded in different play situations with the particular environment used left up to each student.

Malone's second motivational principle is that the activities should contain a variety of challenges and goals that appeal to students of different ability levels and different motivational structures. Also, the goals should change as the student progresses, so that the uncertainty of success is maintained at a small but constant level.

Related to Malone's emphasis on challenge is the element of "curiosity," which is heightened by maintaining the relative amount of information and mystery in the environment. Feedback from the computer regarding student actions should encourage further exploration by providing new questions as well as new understanding. For example, in the computer recreation known as "adventure games," travel through an initially-uncharted maze is accompanied by both further information about the maze and its "dangers" and further questions about how to best proceed. These games may serve as a prototype for the design of educationally useful explorations of a substantive rather than physical maze.

Although Malone's suggestions appeal to many, others may feel that turning learning into play is at best inefficient and at worst destructive of the self-discipline required for future learning. Development of instructional activities incorporating such motivational features should be accompanied by research which evaluates these components empirically.

Computer-Based Drills and Educational Theory

Beyond practical questions about whether computer-based drills will be successfully integrated with the textbook-based curriculum or whether they will be as motivating as alternative drilling methods, and beyond questions about their effects on achievement test scores or their effects on the school budget, the use of computers as drillmasters raises questions of educational philosophy and curriculum theory.

Many people feel, for example, that drill-and-practice activity is already over-used by teachers. After all, repetitive drill is most useful only when the student already knows the relevant principles and procedures, and when the skill to be mastered is merely applying the principles and procedures accurately and speedily. If a child mistakenly thinks he understands how to do the math or grammar exercise, or if he does not know how to attack the problem and is "just guessing," repetitive drill on similar problems will reveal at most the level of understanding the child has--i.e., it serves as a testing instrument, not a teaching aide.

Most drill programs provide only approval for correct answers and mild disapproval for incorrect ones. Howe and DuBoulay (1979) suggest the superiority of providing "information feedback" that tells why the answer is wrong and/or what the correct response should be. Also, such information may be provided at each step in the problem, rather than solely at the final solution. They refer to a study by Tait, Hartley, and Anderson (1973) which found that information feedback produced performance superior to mere reinforcement feedback.

Howe and DuBoulay also find that students become frustrated and begin to respond randomly after some exposure to drill programs that do not provide assistance in learning to solve the problems. Of course, merely giving the correct answer without helping the student understand errors may produce other behavior that is not conducive to learning, such as making any immediate

ERIC

response in order to get the correct answer to appear on the screen.

A related philosophical criticism is that C.A.I. exaggerates an already overgreat emphasis on the learning of facts. Facts, after all, are more easily translated into computer programs than are conceptual ideas. The student, according to this view, is already too often called upon to provide the "right" answer to someone else's question, while little emphasis is placed on encouraging students to ask questions, to organize their ideas, to apply their understanding to new situations, to learn to work productively with their peers to accomplish planned goals, and to learn how to improve their learning skills (Garson, 1980). Drill and practice on the computer, rather than helping students expand their intellects, conditions them to regard the computer as a rather boring tool of the teacher over which they have little control.

However, the critics of drill and practice may have an idealized picture of what kind of instruction is necessary to enrich children's capacities to learn and to relate to others. When pressed, the critics would probably agree that knowledge of spelling is necessary to communicate, that possession of a larger vocabulary is essential to ask appropriate questions, and so on. And the kinds of instruction needed to provide these basic skills are often fairly mundane tasks involving introduction of small pieces of new material, concept formation, practice and review.

Therefore, if "higher order" activities depend on knowing basic language and math skills, using the computer to deliver this instruction can be an alternative to rather than an extension of the time devoted to drill work. The primary question is whether computer-based drill is more educationally effective and more cost-effective than other ways of accomplishing this result.

The factors previously discussed are ingredients of an appropriate answer to this question, but the final outcome will vary for schools that have different financial resources, and the outcome is likely to change over time as the cost of computing continues to decrease. Now, most schools can afford only a limited number of the mass-marketed microcomputers. In addition, the available drill-and-practice software has limited coverage of a given school year's curriculum and in most cases is inadequately integrated with other curriculum materials. Also, we lack concrete information about the educational effectiveness of particular programs available off-the-shelf. Given these circumstances, I believe it is wise for schools to wait for future developments. Developers of drill-and-practice materials must begin to address classroom organizational issues in their system designs. Being practical, they should build these educational software systems for use in computers that will be available in the near future--perhaps for a 1985 generation of $200 handheld computers. Given enough time, the utility of computers to effectively and efficiently increase student achievement in computational arithmetic and language skills may be fully demonstrated. However, if schools invest too much too soon, the disappointments may prevent future products from having a fair opportunity to prove their value.

TUTORIAL COMPUTER-ASSISTED-INSTRUCTION

The distinction between drill-and-practice C.A.I. and tutorial C.A.I. is arbitrary. Drills shade into tutorials as the mode changes from repetitive computer-asked questions and student answers to the presentation of new information displayed on the computer screen; inclusion of more conceptual ideas instead of pure practice of examples; and use of a more complex logic to examine students' patterns of response, analyze errors, and determine

ERIC

appropriate informational feedback and sequence of problem presentation.

Minimal tutorial dialog can be written using a limited number of programming statements that display information and questions, accept the student's response, compare the response with the "right" answer, and branch to a remedial information display or the next sequential information display depending on the appropriateness of the student's answer. At the other extreme, a tutorial program can attempt to model the student's understanding of the subject matter and provide a dialog based on this analysis. It may determine, for example, the optimum point at which to offer hints, what kinds of hints to provide, and how much to emphasize concepts relative to examples. This type of tutorial system is often called "intelligent C.A.I.." A third tutorial model, embodied in one of the two major N.S.F.-funded computer based curriculum projects of the 1970's (TICCIT), is based on the notion that students themselves are best able to determine when help is needed, whether examples or concepts are needed next, and how much to practice any given concept before asking for the next instructional material.

This approach, called "learner control," is used in TICCIT in several ways. Students are provided with displayed "maps" of the course, unit, and lesson terrain, with associated performance status measures. Generalizations and instances are explicitly classed as easy, medium, and hard--both in the expository mode and in the problem-solving mode--and alternative levels of difficulty are available for the student at any time. Help is available to the student in the form of further explanation related to each instance. Keyboard keys are reserved for various functions and options such as RULE, EXAMPLE, PROBLEM, HELP, MAP, and ADVICE. The ADVICE option contains information provided for the student by an advisor who has examined the student's performance. Test files are written for each lesson. The general

ERIC

rule about student choice is that no new instructional material is presented until the student requests it (RULE). Beyond that freedom, the TICCIT system also provides student choice over the alternative sequences.

Alternative designs are possible, however. One could give the computer preemptive or default power over the sequence of presentations within any given lesson or module (Bunderson, 1974); or, the ADVICE key could allow the system to recommend what the student should do next without actually exerting control over the sequence (Merrill, 1980).

A major problem in tutorial programs, as with teaching of any kind, is knowing when to intervene in the student's thoughts and behaviors. Providing information can help the student avoid fruitless guesswork and provide a more efficient "debugging" of the student's theoretical understanding. On the other hand, one may be preventing the student from learning through self-discovery. It seems likely that the more that a student develops an understanding of a problem without being told certain essential facts or relationships, the more thoroughly understood the problem is, and the more easily it may be applied elsewhere. Nevertheless, although it is appealing to give students control over the sequence of problems and "hints" given by the computer, it is not clear that students have a good enough model of what they need to study next or when it would be intellectually productive for them to recieve a "hint."

When tested empirically, learner control has not proved to be more effective at increasing achievement than the more traditional ("adaptive") C.A.I. in which the computer programs choose the sequence of tasks (Merrill, 1980; Grubb, 1977). An argument can still be made, however, that learner control helps students gain skills in learning how to approach new learning tasks, and that these consequences are not measured by the research that has compared various tutorial models. Gagne suggests that the proportion of instruction that should be under learner control is a function of age and grade level (quoted in Hickey, 1974).

Some questions concerning TICCIT tutorials apply to other tutorial models. Tutorials involve a great deal of verbal presentation on the computer display screen. Yet reading vertical material on display screens, particularly through the less expensive video display units used in the computers that schools tend to buy, is not as comfortable for most people as reading from a book. One possibility may be for the tutorial programs to direct students to off-line reading from texts for expository material or even problem presentation. However, if the computer is not determining the sequence of materials (this is determined by the student under learner control principles), one is left only with computerized drill-and-feedback.

Another problem with heavily computerized dialog is that most students are not yet facile with keyboard communications. Gagne feels that having to communicate to the computer through a keyboard and video display is "one of the biggest bottlenecks in the whole business" (quoted in Hickey, 1974). The greater the tutorial's capacity for dialog, the more that typing skills, spelling proficiency, and the ability to follow directions become prerequisites for adequate use of the computer.

The most problematic aspect of tutorial C.A.I., though, is not related to the user's competencies but to the computer program's competencies. Using computers successfully to direct appropriate sequences of stimuli to the student requires a clear understanding of how people acquire skills and knowledge. For computer-based instruction to be effective, the programs must incorporate into their procedures the sequences of material, response-contingent alternatives, user-controlled options, and feedback mechanisms that promote learning for the greatest number of students. Yet, as Ellis points out, "even the best teacher does not know completely what it is that makes him a good teacher" (Ellis, 1974). Therefore, our ability to create computerized interactive procedural environments is limited by what little knowledge we have about optimizing instruction.

Some basic principles of learning are obvious: instructional methods should involve students actively in the learning process; practice is a major element of learning; correct performance should be reinforced; and student motivation is an important affective component that affects the efficiency of cognitive learning.

But knowing these principles of learning does not help a great deal in formulating and developing computer programs that would improve learning effectiveness over traditional methods. Brown (1977) enumerated several practical problems that have to be addressed before fully "intelligent" computer-assisted instruction can be realized: (1) A practical theory of hints. Suppose a student has no idea about how to solve a problem. What information should the computer provide? (2) An algorithm of hints that makes them a function of the skills an individual student has mastered and those he has not: Suppose the student solves part of the problem, but gets stuck. (3) A theory of interference: Suppose a student begins to make errors while solving a problem. Should the computer stop the student as soon as the first error is made, or at what further point? (4) Appropriate limitations on computer intelligence: Intermediate steps of a student's solution may not necessarily be provided to the computer. Yet an intelligent tutor may be able to deduce the intermediate step which caused the error. Should the computer be provided with the same kind of skills?

The teacher's capacity for performing these kinds of tutorial interventions is clear. The computer's is not. The scholarly and programming effort required to improve tutoring abilities of computers could involve lifelong careers in and of itself. The microcomputer programs now being marketed have few of these intelligent features. The practical question now is whether the simplest tutorial programs, combining a presentation of facts, multiple-choice questions, and some response-contingent branching, provide useful instructional advantages.

Simple instructional tutorial programs are available for microcomputers in a variety of subjects from astronomy to zoology. However, most are isolated exercises that cover only a small portion of textbook material. Some of these programs may provide an innovative way to study one portion of the semester's work, but by themselves would hardly justify the purchase of microcomputers as instructional tools.

Even the simple tutorial C.A.I. programs are few in number, perhaps due to the large volume of programming and writing required to produce effective verbal dialog on the computer. Uttal (1969) believes that the most limiting aspect of "canned tutorials" is their dependence on a large dictionary of material. First, the task of entering dialog into the computer specific to particular vocabulary and problem sequences is very time-consuming. Secondly, it is "remarkably tedious" to anticipate all the dialog needed to carry out an instructional conversation based on all combinations of prior and current student response. "This programming bottleneck has been squeezed through only occasionally, and then by an almost brute-force approach which has seen marginally qualified people actually becoming the direct authors of widely distributed teaching materials." (Uttal, 1969; p.2)

Beyond the practical problems, Uttal sees a basic theoretical limitation to producing a "completely adaptive conversation, for it is impossible to conceive of a completely adequate dictionary." He compares this type of tutorial system, which he calls "selective computer teaching machines," with pictographic language systems of the Far East. In both cases, the limitation is the lack of generality in the scheme used to represent concepts.

A similar distinction explains the greater adaptability of computers for use in mathematics, and secondarily in science, than for use in language, social studies, or other instruction. Concepts in math and science are smaller in number and represent greater levels of generality than everyday linguistic discourse and the verbal symbols used in studying subjects such as history and literature.

Uttal's phrase for using more powerful computer programs in subject-matter instruction is "generative C.A.I." Generative tutorial systems attempt to model the behavior of human tutors, in contrast to procedures such as programmed instruction, which model the end result (vocal utterances) of tutor thought processes. "We must always remember that the human tutor does not operate in the fashion of a table scanning device--i.e., he is not a dictionary user. Rather, he is an analyzer and generator who determines what his student's needs are, and then from some general set of rules or heuristics formulates a sentence, a problem, a diagnostic, or a remedial unit" (Uttal, 1969). Uttal felt, in 1969, that only mathematics, the physical sciences, and perhaps music were amenable to tutorial programs. It is not clear that things are substantially different twelve years later.

Thus, tutorial computer-assisted-instruction, however appealing, may not be particularly practical, except for limited use in mathematics or science.

Displays must be highly specific to a particular dialog history in order to present appropriate comments and make appropriate branching decisions. This requires very large programs and time-consuming programming. To keep to a reasonable size, tutorial programs must be limited to "understanding" only a fraction of possible student verbal responses or the program must cover a narrow range of material. The potential value is great, but simply producing more C.A.I. programs will not help realize that potential. We must learn how to produce better programs. As Ellis commented, "James Moody is a musician not because he plays music but because he has the capacity to play music.... It is a variant of this notion to believe that the more we do with computers in education, the greater will become our capacity to computerize education and nowhere is this self-deception greater than it is among the advocates of tutorial C.A.I...." (Ellis, 1974; p.62)

MANAGEMENT OF INSTRUCTION

School districts use computers for other than instructional functions. They may use computers to manage and report student attendance and classroom enrollment, to accomplish payroll and other accounting functions, and to manage materials and supplies including, for example, library books. For these purposes, using microcomputers or combinations of microcomputers and central office computers linked by telephone lines is an efficient and convenient alternative to other automated or manual procedures. This paper will not discuss such uses. However, automating these administrative functions clearly requires fewer computers per school than the number required for computer-assisted-instruction. Schools that purchase a single micro-computer for C.A.I. use may find that administrative use makes more sense at this time. Some programs are beginning to be marketed for administrative uses of Radio Shack and Apple computers with disk storage systems, but I am unfamiliar with their performance.

Another administrative function, the recording and use of student test performance records, relates to the instructional use of microcomputers. Machine-scored standardized achievement tests are used in most school systems to summarize student performance in basic skills and certain applied academic skills. These tests are used mainly to compare the performance of groups of students, to assign individual pupils to classes the following school year, and to identify children who need special instruction. However, most of these tests can be used to designate which particular skill-learning tasks should be targeted for the instruction of which particular students.

Using tests to individualize instructional efforts keyed to identified skill deficiencies is not a common school practice. It requires organizational planning by the teacher, but much research has suggested that this strategy has merit. Microcomputers can be functional appendages to a system that uses test results to select individualized instructional tasks. They can provide individualized testing where the student receives the problems and enters his or her responses at the computer, or they can analyze the answers from paper-and-pencil tests and select subsequent instructional tasks. The former mode is often included with computerized drill-and-practice programs. The "management" functions of such programs give the teacher a summary of student performance and maintain records within the computer so students are given appropriate individualized drills the next time they "log-on" to the computer.

The microcomputer can also be used to manage instruction independently of how students practice their language or math skills. Optical mark-sense scanning machines can be interfaced to microcomputers to provide off-line scoring of answers to individualized worksheet drills which all students can be completing simultaneously. The results can be entered into the microcomputer by way of the optical scanner and analyzed as above. The main advantage of this method of providing individualized stimuli and feedback is that a single computer can accomplish the management for an entire school. The disadvantage is that this application is limited to use with multiple-choice formats, because the scanners can determine the location of marks, but cannot read actual answers. For tests using open-ended numerical or short-answer verbal responses, either the students or a clerical assistant could type the student responses into the microcomputer, which would then continue as above. This procedure is more time-consuming than using a scanner, but it can be used with any type of individualized drills regardless of their answer format.

Few schools use microcomputers to manage instruction independent of a computer-based drill-and-practice program. Several publishers do offer computer programs for test answer entry and scoring. However, I am unaware of any product on the market which includes paper-and-pencil skill worksheets as well as a computer program for entry of student responses, maintenance of student performance history records, and individualized selection of the next sequential learning task. However, given the much more limited computer hardware requirements of this type of drill-and-practice system in comparison to student hands-on C.A.I., I would expect developments in this direction soon.

SIMULATION, MODEL-BUILDING, AND PROBLEM-SOLVING

Drill-and-practice and computer-based tutorials dominate most discussions of instructional uses of the computer primarily because they promise to help educators perform functions which have become central elements in the school curriculum--teaching computational procedures in arithmetic and teaching the components and pieces of written English. Another reason, as we have suggested, is that these learning activities are more easily modeled on the computer than are other aspects of the school curriculum. Thus, computer-based products for schools tend to be developed to address those learning activities rather than to address some others for which computer applications may in fact be more suited.

Licklider (1979) suggests that it is easier for "stupid people" to write factual programs (math drills, multiple choice tutorials) than to write programs that help students understand concepts or perform inductive or deductive logic. It is also easier, he says, to treat people uniformly than to respond to individual learning needs, capacities, and so on. Thus, says Licklider, "a large part of the ... dangers in the application of information technology to education is related to the fact that what is easiest and most profitable to do is, by and large, not at all what should be done in the broadest interest of the society and mankind." It is important, he says, to "exploit" opportunities offered by technology without letting organizations responding to narrower interests "exploit" people's wishful thinking about how technology can be used.

Licklider feels that efforts should be directed toward long-range developmental studies on using computers to foster discovery and the ability to organize ideas, rather than trying to use present inexpensive equipment for the more mundane applications like computer-managed-instruction, drill-and-practice, and question-and-answer tutorials (Licklider, 1979).

Licklider's statement is a reminder that disagreements about the kinds of classroom activities that computers should encourage can easily become a debate about basic educational philosophy. One view favors such practices as clear behavioral objectives, authoritative explanation of concepts and processes, and a great deal of practice on well-chosen exercises. The other perspective asserts that learning a subject thoroughly involves intrinsically motivated activity and internal control over problem selection and method of attack. People's interests in how computers could be and should be used in schools seem to parallel their basic theory of how learning best occurs.

This contrast between educational philosophies need not necessarily translate into parallel attitudes towards drill, simulation, or other uses of the computer. Tutorials or drills can be developed that provide intrinsically motivated activity, student choice, and exploration of the computing environment. However, such development requires a more systematic effort than usually occurs in the rush to be the first to market a particular kind of product. At the same time, laboratory simulation programs can vary from prepackaged "cookbook" curricula to those that allow unanticipated variations in parameters and even changes in the basic model being simulated--all at the option of the student.

Simulations

Simulations and games became popular instructional tools in the late 1960's and early 1970's. Because they require active engagement on the part of students, simulations were considered to be a motivating and enjoyable way to teach facts and concepts. And, because most simulations are close substitutes for actual experience, simulations were seen as particularly useful for teaching about dynamic real-world systems including concrete historical situations, the operation of economic enterprises, and social conflict. Additionally, simulations, like motion pictures, can be used to engage students for further study on a topic. A particularly well-written computer simulation that has been used in this way is "Oregon Trail," a simulation of a family's covered wagon journey to the West in 1847. Written originally as part of Minnesota's statewide library of educational computing materials, the program has been adapted to run on a variety of microcomputers.

Simulations are usually distinguished from "role-playing" games in that they have an explicit set of rules, participants with explicit amounts and types of resources, and, usually, pre-ordained goals and a measure of success in meeting those goals. The explicitness of simulations limits the variety of events that are considered legitimate (within the game) compared to more fluid role-playing games. However, the players' permissable alternative actions are meant to reflect the actual alternatives available in the real world.

Most simulations produced for elementary and secondary classrooms do not involve the computer, although many could profit from at least incidental use of the computer for arithmetic computation and distribution of information to participants. Nevertheless, many of the advantages and disadvantages of using non-computer-based simulation activities in classrooms apply as well to computerized simulations.

Simulations, like some other organizational forms for instruction such as team competition and peer tutoring, usually demand that classroom authority be at least temporarily distributed more widely than usual and that multiple activities be allowed to occur simultaneously. Much of the resistance against using simulations may come from the organizational problems that many teachers believe simulations create. Also, however, research on the learning consequences of simulations has not been uniformly positive. Most studies have found no learning advantages, although student attitudes have shown improvement.

In developing simulations, one design question concerns how complex to make them. The more complex the simulation, the more difficult it is to accomplish learning objectives because the student becomes increasingly unable to isolate the consequences of any given action when so much else is taking place. Elder (1973) suggests employing "progressively developing simulations," simulations either built up by step-by-step accumulation of greater complexity or composed of segmented subsets of problem variables.

There are two basic organizational forms in which computer-based simulations can be used for classroom instruction. Like tutorial and drill uses of the computer, computer simulations can be one-on-one, with each computer terminal controlled by a single student. This structure overcomes objections to the decentralized focus of classroom attention. On the other hand, computer simulations are one application of computer technology in the classroom that does not require that each computer be used by only a single student at one time. Simulations may provide the focus of attention, and be the rule giver, the scorekeeper, and the playing board for an activity that involves many students. In this respect, simulations are a more practical instructional application of computers in classrooms that have a limited number of computer stations.

Simulations may provide students with insights into things that would otherwise be beyond their experience. However, to produce accurate understandings, the simulation model must correspond closely to real-world conditions. Thus, simulation models must specify their boundaries, which limits the extent of discovery available to students. In contrast, actual scientific activity allows the model itself to be subject to inquiry and change.

Model-building

Constructing or modifying a simulation model is called model building.

In model building, a student is asked to specify basic aspects of the model and to judge how well the system corresponds to the real world. In simulations, the model is already formed, and the student uses the model to discover what the system predicts under a given set of initial conditions. Usually the student is asked to assume that the model represented by the simulation is an accurate reflection of the real world (Luehrmann, 1977).

In mathematics and in language, students are often taught "rules" to follow. ("If the numbers when added together total 10 or more, carry the left digit to the next column and record the right digit under the summation.") Rarely are they taught to discover the rule that governs a set of procedures--that is, to build models. Yet this inductive learning may be particularly important for retention of information and for a deeper knowledge of the rule itself.

The computer provides a good vehicle for teaching rule discovery. Rules of a system may be made variously simple or complex; they can be modified to challenge students as they become more skilled; and they may take on a variety of different forms to become more challenging still.

An example of a model-building exercise that has been written as a computer program is called Game X. Game X is a person-against-computer game like checkers, but the player is not told the complete set of legitimate moves. He must determine these from the kinds of moves the computer initiates or makes in response to his own moves. The player can also observe the computer's moves to learn optimal strategies (Goldberg, 1979). This is a general model-building exercise, but more subject-bound examples have been developed.

Simulations and model-building may be advanced and complex tools for use in elementary and secondary schools, but computer programs can be used to help students discover principles and processes on very simple levels--simpler

ERIC

and more concrete than the Game X example discussed above. In this mode, the computer may demonstrate to the whole class or tutor a single student or a small group of students.

Consider the example of using a computer program to demonstrate the isomorphism between graphs and charts and the relationship of both to quantities of objects having different characteristics. A program might display alternative representations of the quantities and also enable the student to modify quantities and observe how this affects their representations as a chart and as a graph (Goldberg, 1979).

As another example, geometric rules can be observed by dynamically changing one aspect of a geometric figure to see how other aspects change or remain constant; i.e., the lines connecting the midpoints of adjacent sides of any quadrilateral always form a parallelogram. The computer could be used to dynamically demonstrate principles and concepts such as these in many ways.

Unfortunately, few programs for using simulated models or for building and testing alternative models have been developed for the microcomputers schools are now buying. Fewer still have sufficient supporting textual material to integrate computer-based activities with a larger teaching unit on a given topic.

The exploitation of the computer's ability to demonstrate consequences of changed conditions and assumptions will require creative effort. We have too few examples of computer-based instructional simulations to reliably judge the value of this approach. Teachers interested in expanding higher-level inferential skills of students need to be exposed to computer-based simulations, so they can contribute ideas and models for new simulations. Only with a sufficient range and variety of examples will we learn whether computer simulations and model building are practical methods for providing instruction for various subject-matters and grade levels.

CURRICULUM-MODIFYING APPLICATIONS OF MICROCOMPUTERS

Simulations and model-building, although removed in most implementations from lock-step instruction and rote learning, are still primarily viewed as methods of teaching math, language, science, social studies and other traditional academic subjects. Other uses of microcomputers in a classroom, though, imply definite changes in the school curriculum. That is, such uses concern not only how children should be taught but also what they should be taught.

The argument for such uses goes something like this: Because the purpose of schooling is to prepare students for the world after schooling, instruction should relate to intellectual activities that will be predominant in society a decade or so ahead. For example, interpersonal communication is increasingly written rather than verbal, and is also increasingly based on typed or machine-retrievable printed text rather than handwritten text. In addition, keyboard communication will be for the forseeable future the primary means of man-machine communication as well. Thus, the teaching of typing skills may be reasonably regarded as a basic academic skill that should be learned as early as such skills become useful in the context of education or extra-curricular life.

One of the major outcomes of knowing how to type is an increased capacity to take advantage of computer-assisted instruction. Almost all communication with desktop computers in the immediate future will be through keyboards and video displays; thus the more rapidly and effortlessly that students can translate what they want to say to the computer into machine-readable communication, the more effective will be the time spent at the computer station.

Several of those who have studied how computers have been used in schools have reported that a major obstacle to their effective use has been the students' relative unfamiliarity with using typewriter-like keyboards for written expression (e.g., Holzman and Glaser, 1977).

Where once the computer was the expensive tool of space science and Fortune 500 corporation financial management, computers and computer-based telecommunications are becoming a part of most white collar jobs and entering the realm of private residential consumption. The computer will be used in these new settings to prepare text for communication and to selectively retrieve information based on individually chosen criteria. Given the speed of advance in computer and communication capabilities, it may be that the major impediment to widespread use of computers in everyday business and personal lives five years from now will be that too few people will have had enough exposure and experience to effectively use this tool.

Anyone faced with a new appliance, whether a food processor or an automobile, needs to be taught how to use it effectively. However, prior preparation in using a similar appliance lowers both the emotional threshold and the cognitive demands required to master the new one. Just as library skills are considered to be a basic subject in the secondary school curriculum today, so might we consider such skills as learning to compose and edit text and learning to manipulate a large data base to retrieve particular pieces of information. Development of prototype curricula in these areas should be a priority in education today.

COMPUTER PROGRAMMING IN THE SCHOOL CURRICULUM

Teaching typing, text-editing, and information retrieval are examples of curriculum changes in which the computer is treated as a tool or appliance--

uninteresting in its own right, but valuable for the information it helps process. Such uses are not qualitative_ ifferent from using the computer to drill students in basic math and language skills. In one view, this is the primary way that computers should be regarded in schools--as tools. An official of Science Research Associates, a company that distributes computer hardware and software to schools, has suggested that microcomputers will have succeeded in schools when they cease to become interesting in their own right--when they become "invisible," so to speak.

An opposing view is that microcomputers deserve to be treated differently from most educational tools. Because computers can serve such a variety of intellectual functions, there is great social utility in helping many students become skilled computer programmers. This viewpoint has been most effectively stated by Arthur Luehrmann, a pioneer in educational computing, in his paper, "Should the Computer Teach the Student, or Vice-Versa?" (Luehrmann, 1972).

In this paper, Luehrmann tells a parable about a society which passed along information solely by oral means. He relates the sequence of events which transpired in that society upon the invention of reading and writing. The new technology was first used in business and government, where the capacities to read and write were most easily translatable into economic profit. Eventually, however, the "vendors of reading and writing" looked to education as a new market to explore. They developed a system where scribes wrote down the words of master teachers, and then professional readers read these writings to students, at a cost far lower than when solely master teachers were employed.

As Luehrmann describes it, this was the "sad ending." However, he provides an alternate ending in which small groups of isolated master teachers began trying to understand the new reading/writing technology themselves and a few tried to teach it to their students. Their philosophical rationale was that "'reading and writing constitute a new and fundamental intellectual resource. To use that resource as a mere delivery system for instruction, but not to give a student instruction in how he might use the resource himself, was the chief failure of the W.A.I. (writing-assisted-instruction) effort,' they said. 'What a loss of opportunity,' they exclaimed, 'if the skill of reading and writing were to be harnessed for the purpose of turning out masses of students who were unable to read and write!'" In Luehrmann's "happy ending," this new skill gradually became more and more acceptable, even to school administrators, some of whom "became competent and imaginative users of the skill (themselves)" (Luehrmann, 1972).

On a less philosophical plane, Hunt and DuBoulay (1979) suggest two rationales for teaching children to program computers. The first is that "construction of computer programs can bring insight into specific aspects of the subject being investigated, especially those aspects concerned with processes." Lipson suggests, similarly, that a useful programming exercise is to have a student write a program that teaches a subject to other students. As with peer tutoring, teaching by writing a computer program helps clarify the subject in the "teacher's" mind (Lipson, 1980).

Hunt and DuBoulay's second rationale for teaching programming to children is that it "encourages them to develop their ability to formulate precise descriptions of how problems should be tackled." In other words, it "brings problem-formulating and problem-solving skills together in a single context." Bell (1978) notes that programming usually produces actions that are part of good problem solving procedures--for example, restating a general problem in a way that procedures for solving the problem can be operationalized.

Also, good programming practice includes testing one's solution with a variety of "inputs" to be sure that a general solution was obtained. Frequently, a programmer will want to generalize the program to make the solution apply to a wider class of problems.

A similar rationale is given by Papert (1981). Programming always requires "debugging" because our initial understanding of a problem is usually incorrect or too imprecise to be correctly articulated on the first try. Papert argues that learning to debug computer programs is a valuable skill that generalizes to attacking other conceptual problems.

Nevertheless, the generalizability of programming skills is still a question that lacks strong empirical evidence, and in at least one instance, programming did not improve other applications of inferential skill (Statz, 1973, cited in Howe and DuBoulay, 1979).

Others believe that, regardless of the tie between computer programming activities and problem formulation and problem-solving skills, it is too difficult to teach most people how to formulate precise representations of problems so they could use programming skills to solve them. People's literacy levels in most subjects appropriate to computer algorithmization may be too low to permit the utility of computer programming to be realized (Osborne, 1977). Of course, if this is true of the general adult population, it is probably true of teachers as well; and if true of teachers, could it be less true of students?

Numerous other reasons, though, have been given for including programming in the school curriculum. Programming involves manipulating a concrete object--the computer keyboard--which is an important mode of learning, particularly for younger children. Programming interactive computers is in many ways a self-teaching operation. It involves engagement with an immediately responsive environment--one gets rapid feedback about the correctness of one's assumptions—and as such it is motivating intellectual activity. Programming allows for exploratory behavior in an environment with clear boundaries--thus it allows students to exercise a degree of authority which they generally lack; at the same time there is a clear focus and goal to target student attention. Finally, the societal demand for people with programming skills will continue to grow, regardless of the growth of "program-writing programs." Although there will be automation and standardization in computer software, the number and variety of computer applications and the customization required for individual circumstances will continue to increase.

Still, the question remains whether programming is a viable subject for the pre-collegiate curriculum. Of all academic subjects, mathematics most closely parallels the mental activities of computer programming--but a great many people have difficulty learning mathematics. Thus Osborne (1977) could be correct--too few people have the skills required to formulate problems in a sufficiently rigorous way to write computer programs successfully.

An M.I.T. mathematician, Seymour Papert, offers an opposite notion: people's innate mathematical abilities are not too weak, but the way people have been taught mathematics is at fault. Papert feels that teaching programming--in a way that focuses on solving problems using an easily understood computer language--is a primary mechanism for correcting mathematical illiteracy.

"The belief that only a few people are mathematically minded is a truism in our culture and a cornerstone of our educational system. It is therefore sobering to reflect on the flimsiness of our reasons for believing it. In fact, the only evidence is crass empiricism: look around and you will see that most people are very poor at mathematics. But look around and see how poor most Americans are at speaking French. Does anyone draw the conclusion that most Americans are 'not French-minded?'--that they are not capable of learning French? Of course not! We all know that these same people would have learned to speak French perfectly well had they grown up in France. If there is any question of lack of aptitude, the aptitude they lack is not for French as such but for learning French in schools.

"Could the same be true of mathematics? Could there be a place, a 'mathland,' which is to mathematics as France is to French, where children would learn to speak mathematics as easily and as successfully as they learn to speak their native dialect?" (Papert, 1980)

Papert believes that the answer is "Yes." However, he does not think that improved mathematical literacy will result from students gaining experience in programming the currently popular microcomputers. Papert--and others--feel that BASIC, the standard programming language of most microcomputers, is not particularly useful for developing mathematical understanding.

Programming in "BASIC"

Teaching programming using BASIC has a number of problems. Although it is easier to learn to write short BASIC programs than to learn to write simple programs in most other computer languages, complex BASIC programs are not easily understood by others, nor even by the program's author after a lapse in time. This makes them difficult to correct or modify and difficult to learn from. Also, BASIC does not encourage "modular" programming--dividing logical tasks into small, workable subtasks which, like customized building blocks, can be combined in different ways as the need for their function arises. Thirdly, BASIC does not allow manipulating aggregates of data but only individual elements; therefore, higher-order processes must be subordinated to particular details. Finally, like most current programming languages, BASIC does not allow easy simulation of simultaneous activity by different objects. It is a sequential, procedural language.

To overcome these disadvantages, computer scientists at Papert's M.I.T. lab, at the Xerox Palo Alto Research Center, and at other places are developing programming languages which are more easily taught to children and adolescents and which could produce greater mathematical fluency in a wider range of students. These languages, such as Papert's LOGO and Xerox's Smalltalk, do seem to be more appropriate for teaching, although few people outside of the development laboratories have used them in classroom situations.

LOGO and Smalltalk will become available for use in consumer devices over the next several years. However, because they require more computer hardware capabilities than languages such as BASIC, it is likely to be some time until low-priced computers include these languages as integral components. For the forseeable future, teaching students to program will continue to mean using BASIC.

BASIC does have advantages. Although major conceptual breakthroughs in mathematical understanding may be elusive and programming large problems in BASIC may be unwieldy, learning simple programming logic with BASIC is quite easy. Compared to other current programming languages, BASIC has much of the flexibility and immediate responsiveness of spoken language, which makes it attractive for teaching children (Luehrmann, 1981).

Also, BASIC has been implemented on a variety of small computers in dialects that are similar to each other. Thus the knowledge gained is reasonably transportable. The diffusion of programming skills in teachers

ERIC

can be more esaily accomplished than if different teachers were taught different programming languages. Because it can be implemented on computers with limited capabilities, BASIC is likely to remain a standard in education--where the dollar expense for hardware is a prime criterion.

Thus, in spite of theoretical advantages of languages that are more structured or that enable greater growth of conceptual understanding, BASIC's practical advantages make it likely to remain the single option open to schools that are considering adding computer programming to their curriculum.

How and When?

How and when should programming be taught in schools? Many who have worked closely with small groups of children in rather special circumstances believe that children in grades K to 4 can learn rudimentary programming operations, particularly in special languages like LOGO in which almost all operations are built up from a few primitive operations. However, programming does require some abstract thought, and most people have assumed that elementary school children lack both an interest in and a capacity for the mental gymnastics involved.

Kay (1977) notes a general improvement at about age 11 or 12 in a child's ability to "plan general structures and to devise comprehensive computer tools." And, in one report, 11-year-old children were able to write their own C.A.I. drills and instructional games, but only after a good deal of effort by the instructor to provide examples of the kinds of things the computer could do. When asked to develop a long program incorporating most of the programming instructions learned through short exercises, the children needed a lot of guidance (Holzman and Glaser, 1977).

Students in grades 6 to 8 seem, on the average, sufficiently mature to understand the abstractions involved in programming. They are less in need of having to manipulate concrete objects in order to understand something. Yet they are still young enough to have largely culture-accepting views of their environment and thus are amenable to exposure to this completely new learning activity.

Older students seem to separate into two groups--a small proportion who are personally interested in programming activities and a larger proportion who view the computer as just another subject in the curriculum. For example, Piele (1979) reported that students in a sixth grade class given an opportunity to learn to program were much more enthusiastic than students at a neighboring high school.

Other factors should influence when to introduce computer programming into the curriculum. First, students should have some typing skills. Particularly if there are few computers, inefficiency at the terminal would interfere with the instruction.

Secondly, among older students, computer programming is more often selected and enjoyed by boys than girls. Although there is no systematic data on this, I would estimate that if we were to divide students into those who show particular interest in learning how to program and those who do not, boys would constitute about 1/2 of the "interested" group in the first grade, about 2/3 of the "interested" students in the sixth grade, and about 4/5 in the ninth grade. If computer programming is introduced before it becomes socially linked with the male gender, it is likely to reduce the effect of peer and cultural norms on the sex-distribution of eventual programming skills.

Finally, current conditions in schools and in computing must be considered. Although LOGO may be a language which young children can learn, BASIC will remain the most widely available language for many years. Also, because classroom teachers will do the teaching, one must consider the ease of

ERIC

recruiting current teachers into instruction in programming. Most instruction will initially be by secondary school math and science teachers, who have already shown the greatest interest.

These factors indicate that computer programming, along with typing, should be introduced into the curriculum in the first year of middle school or junior high school.

Learning programming requires a mixture of rote learning of programming syntax and semantic knowledge of programming principles taught in a sequence appropriate to the student's prior understanding (Shneiderman, 1977). This suggests a traditional instructional approach: A certain order of topics, skills, and computer language commands is decided upon. Each is introduced for the whole class in the same order. Students practice the same set of lesson-geared examples in order to master the concepts. And concept-introduction, examples, and practice are repeated until each unit of the curriculum guide has been covered.

Watt (1981) believes that programming should follow a different model of teaching--what he calls the "shop class model"--in which the sequence of instructional topics will vary from child to child and no doubt from those of a standard textbook. The teacher's role is to present new material based not only on the prior concepts that the child has learned, but also on the child's goals for using the programming tools. In shop classes, a great diversity of activity occurs because each child works on projects requiring different combinations of tools. So programming should be taught, Watt feels, by allowing each child to pursue an independent programming project and use the programming language as an inventory of tools. Of course, this requires that the teacher know the programming language well and know what kinds of concepts are "do-able" by students at different levels of understanding.

Nevertheless, it remains an empirical question whether instruction in programming is better learned using a more individualized sequence of concepts and material than that which is more typically employed in other subject areas. Research should be undertaken to determine empirically what mixture of traditional instructional design and Watt's shop class model is best.

SUMMARY OF EDUCATIONAL APPLICATIONS OF MICROCOMPUTERS

In the previous several sections, we examined the major instructional uses of microcomputers in elementary and secondary schools. We have seen that the most frequently produced computer product--and the easiest to produce--is one that enables the computer to give structured problems to students on an individualized basis, to record student responses for later summarization for teacher review, to provide immediate feedback to students, and to route students to problems requiring greater or lesser skills depending on student performance. These "drill and practice" programs constitute the major offering to date of microcomputers to the basic curriculum of today's schools.

Instructional student-computer dialog which uses the computer to present facts and ideas to students, to evaluate student responses, and to route students to remedial explanations based on an analysis of the pattern of errors extends the drill-and-practice idea. These dialogs or "tutorials" vary from simple exchanges involving relatively little analysis by the computer to complex programs that analyze and diagnose error patterns in student responses. But most involve exchanges of verbal rather than mathematical concepts and suffer from an overly simplistic notion of verbal discourse. These programs tend to be less successful than drills because they are unable to semantically decode a sufficiently wide range of human responses.

Computer-managed-instruction provides a way for the teacher to identify the particular concept deficiencies of individual students. This can accompany computer-based drill and tutorial programs or it can be built into paper-and-pencil learning activities, an alternative which requires fewer computer resources. The latter approach loses any motivational and learning advantages that might be gained from student hands-on contact with a responsive computer program, but it produces informational feedback to the teacher (and student) about precise academic strengths and weaknesses of individual students that can be used to individualize instructional treatments.

Simulations and model-building are computer-related products that may be less broadly applicable than drills and tutorials and which require greater programming effort than most drills and most practical tutorial programs. Nevertheless, they make use of computer capabilities that are difficult to duplicate by other media. In contrast, computerized drills and tutorials merely replace one delivery system with another. Furthermore, as long as computer hardware remains expensive compared to other instructional aids, simulations and examination of models used in a demonstration mode can simultaneously serve much larger numbers of students than those applications requiring one-on-one interaction.

Computers may also be used to extend the curriculum to make schooling more relevant to the cultural requirements and opportunities that students will face in coming decades. The mere presence of the computer illustrates how important typing competence will be, and provides an opportunity to extend instruction in that subject and in allied areas such as computerized information retrieval.

We also discussed the appropriateness of computer programming in the pre-collegiate curriculum. It is my guess that of all the ways in which microcomputers have been used in schools, teaching programming has been the most common and the most successful. (The national survey which we will undertake during the next few months will provide much needed information about computer uses in schools.) However, it is still too early to know all the details about how to teach programming to students--at what grade level, with what prior mathematical understanding, using what types of computer languages, and using what teaching methods.

SOCIAL ORGANIZATION OF COMPUTER USE IN SCHOOLS

Throughout most of this paper we have concentrated on particular instructional uses of computers in schools. Now we turn to issues related to how the use of computers may be organized irrespective of the instructional operations which they may be performing.

Regardless of how capable microcomputers may be for giving students practice with drills, engaging students in a useful dialog, demonstrating the properties of systems through a simulation, or helping students obtain useful keyboarding or programming systems, their theoretical capabilities are irrelevant if they cannot be used effectively by teachers and other school personnel in the ongoing activities of existing educational institutions.

Foremost among the problems to be overcome is the contrast between the capacity of a computer to interact profitably with a single student and the group-based organizational structure of schools. Although schools may vary somewhat from the common pattern, most instruction is provided to students in spacially segregated groups of 25 to 35 students directed by a single adult in an enclosed area. In secondary schools, and often in elementary schools, the composition and location of these groups changes hour by hour as students move on to study different subjects with different teachers.

In this context, three things about microcomputers stand out: Although microcomputers are relatively compact and portable compared to previous computers, they are still "machines" to which students have only temporary access; secondly, their predominant mode of use is with a single individual at any one time; and finally, for some time to come they will be costly compared to other classroom learning materials and will be easily outnumbered by students present.

Centralized vs. Classroom Placement

Computers, thus, constitute a limited resource. Schools must decide not only what material students should learn on the computer, but also which students should have access to this resource and how the equipment should be placed within the school. There are three ways that computer equipment might be allocated: (1) distributed to individual classrooms under individual teacher control, more or less indefinitely; (2) housed in a central warehouse facility, as audiovisual equipment is now, but directed to individual classrooms as needed or requested; and (3) located and used in a central facility, such as a math lab room or the school library, under the supervision of a staff member responsible for the activities in that room.

Several factors argue for centralization of placement and authority over use. Given the expense of the hardware, schools naturally desire to maximize the number of minutes of the school day during which each computer is in use. Placing a single individual in charge of computers may produce more efficient use of the equipment than if each teacher tries to oversee use of the machine(s) in her room while also managing instruction for an entire classroom. Secondly, centralization could allow entire classrooms or classroom groups to receive simultaneous instruction using the computer, leaving teachers free to handle other instruction for individual students or groups of students.

Computers, like typewriters and books, are essentially intellectual environments for the use by a single person at any one time. Classrooms are primarily group interaction systems, often with a single focus of attention for all individuals. The management of individual activities is often best done in a setting, like a laboratory, designed around the idea that everyone is conducting individual activities. In contrast, when a small number of computers are located in a classroom, it becomes difficult to manage group instruction for some children while other children alternate between listening to the teacher's presentation and conducting independent study at a computer.

Centralization is also suggested by concerns over the security and care of the equipment and the expertise required to take advantage of more sophisticated data management opportunities using the computer. However, these points must be regarded as "tendencies," because individual differences in implementation and personnel may be more important than placement of the computer in determining these outcomes.

In addition, there are several good reasons to support the decentralized allocation of computers. First, the most successful uses of computers may be those developed and managed by particular teachers who, for one reason or other, have become experts or enthusiasts for their use. It may be best to leave the computers in the hands of those teachers who have expressed interest in using them, regardless of possible distributive inequities within the school.

Also, centralized allocation may not replace ideosyncratic expertise and enthusiasm with greater efficiency, if it is accompanied by centralized authority over how the computer is used. This may produce a more standardized computer-based curriculum and less clear relevancy of computer activities to classroom activities.

Sheingold (1981) observed one instance in which there was an overall relationship between the classroom math work and the computer drills in the resource center, but there was no matching of the exercises performed on the computer and the instruction being given in class. Individual differences tended to be neglected due to the limitations of the computer program and the segregation between the supervision over computer work and observation of the students' performances in the classroom.

Davis, head of the Plato system at the University of Illinois, believes that computing experiences SHOULD be an integral part of the classroom experience, and that the presence of the computer in the classroom will encourage more teachers to become exposed to and experienced with computers.

The choice of how microcomputers are distributed in the school may not be independent of the decisions about their use. Although this proposition needs further empirical elaboration, a strong relationship appears to exist between placement and predominant use (c.f., Sheingold, 1981). Where computers are under the control of classroom teachers--particularly teachers who have some interest in computing themselves--there is likely to be little talk of "improving learning outcomes." Instead, the focus may be on curricular enhancement and exposure to computers as part of general education. In contrast, centralization of computer facilities is often accomplished in schools emphasizing C.A.I. dril . computer-management of individualized drill, and record-keeping. Nevertheless, the location/authority question and the function question are still logically independent.

Regardless of where the computer is housed and under whose authority it is placed, attention must be paid to keeping the equipment in use--rather than allowing it to remain idle or in need of repair. This would not normally be a high priority among most teachers, whose orientation is toward managing a classroom of children and covering instructional material in the curriculum. Optimizing the use of expensive equipment is not central to most teachers' concerns; thus, investing school system money on computer equipment almost demands that supervisory personnel be diverted or hired to support teachers using computers and to oversee their use. There are always questions about using equipment and specific programs, modifying programs to fit circumstances, equipment malfunctions, and so on. Depending on the quantity of computer equipment, school systems may want to locate supervisory people in the school buildings or in a central office, or may engage support services on a contract or consultant basis. Kenneth Brumbaugh, who directs the Minnesota Educational Computing Consortium (Minnesota's statewide support service organization), suggests that providing support for teachers using computers in their classroom is as important as providing them with software to use.

Besides One-On-One

Most computer programs written for the classroom market assume that each computer is being used by a single child at any one time. As we have seen, this assumption, when translated into products, severely limits the number of children in most classrooms that can be served by available equipment. The assumption about one-to-one use has origins both in the "programmed self-instruction" historical development of educational applications of computers and in the practical limitations of most current equipment--a small video display screen and a single keyboard. Although the practical limitations are not inherent--imagine, for example, a large video display and several keyboards connected to the same microcomputer--they do affect how most people think about program development for classroom use.

On the other hand, it is clearly possible to develop learning activities, including game-like drills and simulations, that involve pairs or several students simultaneously. In addition to providing an environment in which more children can be occupied in a directed learning activity, such programs would also address the preferences that most children and adolescents have for interpersonal rather than individualized learning situations. Writing programs that are interpersonally interactive as well as person/computer interactive certainly requires more creativity than writing drills. However, for now, they may be more functional for typical classroom learning environments. The programs could involve students challenging one another with problems for which the computer is judge (regarding issues of "fairness" as well as "right" and "wrong") and official scorer; interactive games in which the computer is the "playing board"; and simulations in which students take on different roles or work together to solve problems.

Social interaction is by far the most common mode of learning in school, and computer programs could build on this fact. Levin (1981) reported on the use of computers by children in an informal educational setting. Despite an abundance of computer equipment and activities designed for individuals, the most common social arrangement observed was groups of children gathered around a single computer working together at a computer-related task, leaving most other machines unoccupied.

Another alternative to individualized and small-group work at the computer is use of the computer for whole-class demonstration purposes. The major drawback here is the size of the computer display. Nevertheless, if single-computer classrooms continue to be prevalent at least in the short run, it would be worthwhile to develop innovative ways to use the computer in a demonstration mode despite the visual difficulties.

Classroom Microcomputers and Home and Other Influences

Schools are not considering the use of microcomputers in the classroom in a cultural vacuum. Their involvement grows out of the same social developments that are affecting other institutions in society, where small computers are increasingly being bought and used. Many students are gaining exposure to microcomputers and microcomputer-based video games outside of school. Presumably, this exposure influences how they respond to assignments to use the computer in particular ways.

Computers, like few other appliances, have a wide variety of uses. Sometimes when an object is used for different purposes, such as a table used for both eating and writing, the mental baggage associated with one use distracts from efficient use of the object in other ways. This is why teachers often suggest that parents set aside a special place in the home for their child to do homework.

The multipurpose nature of computers raises a similar problem for the design of computer environments for classrooms. The computers that schools buy for their classrooms are sold in the consumer market, often as game-playing machines. The similarity of these computers to electronic video games raises the question of whether these alternate conceptions of use ("I could be playing Space Wars; instead I have to solve these dumb arithmetic problems.") interferes with the implementation of instructional programs on the same equipment.

Will students who are accustomed to interacting with responsive video devices for non-intellectual recreation resist using the computer for drills more than students who have not previously been exposed to any computer use?

What about students who have gained some programming skills--will they be less willing to use computers in the classroom as a teaching machine over which they have little control? One is tempted to answer these questions in the affirmative.

However, the effects may not be totally one-sided. Students with prior exposure to similar instruments seem to be less apprehensive about using computers in the first place. And although they may prefer to "play games" on the computer, most children do understand that at school it is often necessary to engage in intrinsically less interesting activities than one would choose on one's own. Program developers writing for a recreational or home market may need to pay more attention to infusing their learning materials with motivators like graphics, color, and game-like environments than do producers of a similar product expected to be used under a teacher's direction.

Some Possible Undesirable Consequences of Computers in Schools

Computer-based learning in schools may also have consequences beyond the subject-matter learning that takes place. A primary concern of teachers (Lichtman, 1979) and C.A.I. researchers (Seltzer, 1971) is that computer-based learning can be isolating and can have deleterious effects on the interpersonal social skills of students. An emphasis on using the computer for individualized drills would certainly make such a hypothesis plausible. Alternatives that we discussed earlier--designing computer-based activities for small groups, using the computer in a classroom demonstration mode, and using networks of interacting computers--would minimize such outcomes as well as make such experiences more enjoyable (Lipson, n.d.).

There are also claims that computer-based experiences leave people intellectually less able than more social experiences. The highly structured, limited discourse permitted by computers may be a poor substitute for discourse that is more broadly knowledgeable, combining reasoned argument about complex ideas informed by information retrieved from human minds rather than stored on magnetic tape. These fears may reflect a general fear that quantitative style will overpower and suppress qualitative aspects of classroom learning (Friedman, 1980).

In addition, computer-based learning environments, using visual, pictorial, and graphic media and often including auditory symbols, suggest a different orientation towards learning than a more traditional emphasis on the written word. A librarian whose establishment has been recently augmented by several computers suggested that although the exposure to computer skills was no doubt beneficial in some ways, she doubted strongly that it would carry over into increased use of the books in her library (Yasaki, 1981).

Whether these more multisensory media increase intellectual skills is still debatable. A recent NYU study found that people who watched the much-praised "Ascent of Man" series on public television could not remember much of the cognitive information that had been conveyed. In contrast, proponents of computer-based learning point to the active nature of work at a computer terminal, as opposed to passive television watching, and suggest that the apparent similarity between computer-centered activities and other technologically advanced media should not be accepted at face value.

Another social issue related to computer use in schools concerns the social distribution of computer-related skills across population subgroups. Private ownership of microcomputers is strongly tied to social class and family values. It has been argued that schools need to have microcomputers in order to reduce the private disparities between children in learning

ERIC

computing skills outside of school. However, to date, it appears that schools that are investing in microcomputers are disproportionately those in the wealthier school districts.[3] A recent market survey shows a penetration of 30% in schools with a very low Title I eligibility rate compared to a penetration rate of 12% in schools with substantial poverty (Market Data Retrieval, 1981). Insofar as microcomputers present vocationally, intellectually, and culturally useful opportunities to students, current patterns of adoption are likely to parallel rather than counterbalance private distributional inequities.

Within each school, similar questions of distribution, access, and types of use arise. Some schools tend to allow only the brightest and most motivated students to use the limited number of microcomputers. In other circumstances, drill-and-practice uses may be channelled to the lower-achieving students in the school, while the better-prepared students use the computer to create programs of their own choosing. This may be "optimal" use of the computer facilities in one sense, but such divisions may exacerbate the variations in attained academic competencies between the initially slow and the initially high-achieving students in the school.

For each of the problems that may result when microcomputers are introduced into the school's ongoing educational structure, systematic research could help discover their incidence and severity and the conditions under which the problem is minimized. However, in the absense of research,

---

[3]One non-profit organization, People's Computer Company, received federal funding to install computers in the public libraries of a municipality and to develop activities, such as a parent-child programming course, that bring computer literacy to an entire population. However, it is not insignificant that the community selected for this demonstration is one of the wealthiest, most computer-literate, suburban areas in the U.S., located adjacent to Stanford University and one of the major centers of microcomputer development.

ERIC

those who implement various uses of microcomputers in different educational environments should be aware of these problems and should share with other educators how they deal with them.

FROM REALITIES TO DREAMS FULFILLED--THE NEXT STEPS

This paper has conveyed contrasting feelings. On the one hand, there is the excitement felt by those who perceive the microcomputer to be a harbinger of monumental change in the capacity of information machines to affect the method and content of students' education. On the other hand, there is a strong feeling that applying current microcomputer technology with limited equipment and without forethought about integration with traditional classroom instruction will produce disappointments and poorly utilized resources.

However, although it may not be possible to implement computer-based instruction in every classroom today, many activities can still be undertaken to improve the way that computers affect tomorrow's educational process. A follow-up paper will discuss the major steps that should be taken now by researchers, product developers, and school system personnel. This second paper will treat the following points at greater length.

Researchers

The major contribution of research is information about processes and consequences. The claims made in this paper are based primarily on rational argument. individual observation, and the experiences of "expert witnesses" such as practitioners and developers. For reliable decision-making by schools and by product developers, more systematic information is necessary. We need to develop an unbiased and representative body of information about how schools decide to obtain and use microcomputers and other technological learning tools, how they use them, and the effects that their use has on students and on the social organization of the school.

As a first step in this effort, the Johns Hopkins Center for Social Organization of Schools is surveying about 1,500 public and non-public schools to learn how schools that are using computers in any one way differ from those that are using them in other ways and from those that are not using microcomputers at all. Also, the survey will identify factors that are related to particularly successful implementations of microcomputers--in the sense of maximizing their use and their impact on students and the school. Finally, the survey will provide a more factual basis for many of the suggestions expressed in this paper concerning the problems of social organization faced by schools that are using a limited number of microcomputers in the context of classroom-based group instruction of students.

Something which this survey cannot do, but which should be done by any group implementing computers in their instructional program, is to design the implementation so comparisons can be made between different groups of students and teachers given varying "exposures" to microcomputers. Where possible, these exposures should include randomized assignments to "treatments" so that the effects of self-selection can be reduced.

Developers

Although the quest for financial reimbursement for intellectual productivity is a practical consideration, it is important to encourage longer-term developmental efforts than those that provide immediately usable products. Until such time as schools can buy enough microcomputer equipment so that a given application can be easily implemented in classroom instruction, efforts should go into improving the quality, depth, and comprehensiveness of computer-based materials and their integration with traditional instructional materials.

Specific programming details can be worked out at the end of the development period to conform with the characteristics of widely used microcomputers at that time. In contrast, the logical structures necessary to make particular types of learning possible are largely independent of the particular computer used and partially independent of the content, and thus deserve first consideration. Organization of content comes second; and features particular to the make and brand of the computer used in the classroom come last.

Thus efforts should be directed toward developing ways for computer programs to provide instructional assistance; developing bodies of instructional content and methods of presentation; and developing algorithms for sequencing instruction, providing feedback and assistance to students, and organizing the information to be provided to the teacher. A rough guideline of "5 years to market" and technological sophistication equal to today's $200,000 multi-user systems should be assumed.

Simultaneously with this long-range development, efforts to create more immediately usable products should focus on how to make a very few microcomputers functional for a classroom of children: activities for several children at one time at one computer, demonstration-mode simulations, computer management of individualized instructional assignments, and development of materials for teaching programming to teachers as well as to students.

Schools

It is important that schools not be unwitting victims of the enthusiasm of amateur computerists or the aggressive marketing of producers of computer-

ERIC

related materials. At the same time, schools should encourage their staff members to become more knowledgeable about using microcomputers and to seriously evaluate their worth for various functions.

School systems should seek to develop computer literacy in as many staff members as possible, particularly among staff librarians and among secondary school teachers in math, science, English, and business. Teachers with strong disinclinations and those with strong inclinations should be excluded or included as they choose. By computer literacy, we mean familiarity with the variety of instruction-related tasks that microcomputers can be expected to have now or in the near future; and experience in using microcomputers for text preparation and editing, test scoring, and packaged instructional programs. For many teachers, computer literacy should also include acquiring the ability to write BASIC language programs on existing microcomputers and to teach programming.

Staff education should include the policy-makers who will have the responsibility for making purchasing decisions. There is the danger, though, that exposure to computers tends to increase the demand for purchases of computer equipment for classroom use, including purchases which may not yet be wise for schools to make. Nevertheless, widespread staff understanding of computers is a prerequisite for policy discussions--whether at central office or building levels--that should precede any decisions to purchase equipment for instructional use.

The policy discussions are essential and should focus on (1) appropriate uses of computers for assisting traditional and newly emergent instructional goals, and (2) organizational issues regarding the computers' use. Wise policy would refrain from making computer purchases until their function and the manner of their use has been clarified and the plan seems plausible considering (1) the availability and quality of any software and other educational materials that are needed; and (2) the constraints of using individually-oriented learning tools in the context of a group-based instructional organization.

Not to preclude other conclusions resulting from such policy discussions, it is my perception that results of this kind of analysis at present would include decisions something like the following:

At a secondary school, a decision is made to defer purchase of currently-marketed microcomputers until the school can afford to purchase at least 8 units, each with disk storage and sharing one printer among them (an investment of approximately $12,000), and when one staff member--the initial proponent of obtaining computers--has prepared a curriculum in computer programming that incorporates an approved textbook with an appropriate reading level and that uses a programming language essentially identical to the language used in the machines purchased. Waiting until 8 microcomputers could be afforded would ensure that when the class was finally held, it would not be held back by an inadequate ratio of microcomputers to students.

At an elementary school, a hard-headed analysis concludes that microcomputers should not be purchased until at least one of the following activities could be accomplished using off-the-shelf computers, computer programs, and teaching materials:

(1) When, by using an inexpensive set of curriculum-complete workbooks and tests and optically scanned answer sheets feeding into a single microcomputer, the teacher can provide individualized evaluation and individualized sequencing of assignments. The computer programs for analyzing test results must be supplied along with workbooks and answer-recording sheets.

(2) When a variety of computer programs are available to help children explore various capabilities of the computer including sound, color graphics design, and typing and text editing. These are to be used for enrichment exercises for students who have already mastered work for which the remainder of the class requires additional time and teacher attention. This use requires fewer machines, perhaps five or six, centrally located under the direction of a trained staff member such as the librarian. The programs must come with associated teaching materials that provide starting points for student exploration.

(3) When tutoring and remediation in a variety of math and language skills--not simply computer-based drills--can be provided to most of a classroom at one time while the teacher works with small groups of students who have fallen behind the group-paced instruction. Twenty or so individual microcomputers linked for efficient information and program storage would be needed as well as a person who is trained in managing such a facility and who is familiar with the curriculum presented on the computers. This investment, most people at this school believe, is unwise for the forseeable future and unsupported by any set of computer programs on the market.

These are hypothetical policy choices, and many schools will draw different conclusions in essentially the same environment. Microcomputers still have a largely emotional following, and feelings run strong regarding their benefits and their costs. Decisions about their use should be based upon as much information and as much participation as possible.

A Final Word

The advances being made during this decade in the capacity of electronic media to store, retrieve, and process intellectual information at a steadily decreasing cost is one of the more exciting trends in an often-discouraging world. Someday, schools may be able to use the fruits of this technological advance to help children and adolescents attain greater academic competencies and skills than the generations before them. However, it will not help for us to uncritically accept every "computer-based" anything that comes to market. We must think clearly about how we want our children's education to improve, what computers can do to help, how that assistance can, in fact, be accomplished, and whether any of this is affordable. Through appropriate research, well-organized strategies of educational program development, and careful policy-making and staff development by school systems, we may be able to make today's dreams about computers and kids into tomorrow's realities.

REFERENCES

Banet, Bernard (1978) "Computers and early learning," Creative Computing. 4:90-95 (Sept./Oct.)

Bell, Frederick H. (1978) "Can computers really improve school mathematics?" Mathematics Teacher 71:428-433 (May).

Brown, John Seely (1977) "Uses of artificial intelligence and advanced computer technology in education," in Computers and Communication, Robert J. Seidel and Martin Rubin, eds., N.Y.: Academic Press.

Brudner, Harvey J. (1981) "The primary and secondary school markets," presented at the National Technology-Assisted Learning Industry Conference, Chicago, February.

Bunderson, C. Victor (1974) "The design and production of learner-controlled courseware for the TICCIT system: a progress report, " International Journal of Man-Machine Studies. 6: 479-491.

Educational Testing Service (1979) "Taking a long, hard look at C.A.I.," Educational Testing Service, December.

Elder, Charles D. (1973) "Problems in the structure and use of educational simulations," Sociology of Education. 46: 335-354 (Summer).

Ellis, Allan B. (1974) The Use and Misuse of Computers in Education. N.Y.: McGraw-Hill Book Co.

Friedman, Edward A. (1981) "Technology and humanism--are they compatible?" paper presented at the Technology and Education Conference, George Washington University, January 28.

Garson, James (1980) "The case against multiple choice tests," The Computing Teacher. 7: 29-34.

Goldberg, Adele (1979) "Educational uses of a dynabook," Computers and Education. 3: 247-266.

Grubb, Ralph E. (1977) "Student control: exploration in C.A.I.," in Computers and Communication, Robert J. Seidel and Martin Rubin, eds., N.Y.: Academic Press.

Hickey, Albert E. (1974) "Research guidelines for computer-assisted instruction," Albert E. Hickey Associates, Newburyport, MA.

Holzman, Thomas G. and Glaser, Robert (1977) "Developing computer literacy in children: some observations and suggestions," Educational Technology. 17: 5-11 (August).

ERIC

Howe, J.A.M. and DuBoulay, B. (1979) "Microprocessor assisted learning: turning the clock back?" Programmed Learning and Educational Technology. 16: 240-246 (August).

Jamison, Dean; Suppes, Patrick; and Wells, Stuart (1974) "The effectiveness of alternative instructional media: a survey," Review of Educational Research. 44.

Kay, Alan C. (1977) "Microelectronics and the personal computer," Scientific American. p. 231-244 (September).

Levin, James A. (1981) "Computers in non-school settings: implications for education." Laboratory of Comparative Human Cognition, University of California, San Diego. (January).

Lichtman, David (1979) "Survey of educators' attitudes towards computers," Creative Computing 5: 48-50 (January).

Licklider, J.C.R. (1979) "Impact of information technology on education in science and technology," in Technology in Science Education: The Next Ten Years: Perspectives and Recommendations, National Science Foundation. pp. 1-10.

Lipson, Joseph I. (n.d.) "What educational computer systems should be." National Science Foundation, Science Education Development and Research Division.

----- (1980) "Technology in science education: the next 10 years," Computer 7: 23-28 (July).

Luehrmann, Arthur (1972) "Should the computer teach the student, or vice-versa?" in Proceedings of the Spring Joint Computer Conference. Washington, D.C.: Association for Information Processing Systems.

----- (1977) "Intelligent video disc systems--implications for education," in Computers and Communication, Robert J. Seidel and Martin Rubin, eds., N.Y.: Academic Press.

----- (1981) "Child-appropriate computer languages." Paper presented at Conference on Child-Appropriate Computing, Teachers College, Columbia University, (April 30-May 1).

Malone, Thomas W. (1980) "What makes things fun to learn? a study of intrinsically motivating computer games." Xerox Palo Alto Research Center, Cognitive and Instructional Sciences Group, (August).

Market Data Retrieval, Inc. (1981) "Identifying and getting your share of the school market for microcomputers." Market Data Retrieval, Inc. (October).

Melmed, Arthur (1980) "The acceptance of C.A.I. in the educational field and its probable use in elementary and secondary education in the next decade." Paper presented at the Third International Learning Technologies Congress and Exposition, Washington, D.C. (February 25-27).

Merrill, M. David (1980) "Learner control in computer-based learning," Computers and Education 4: 77-95.

Morgan, Robert M. (1978) "Educational technology--adolescence to adulthood," Educational Communication and Technology Journal. 26: 142-185 (Summer).

National Center for Education Statistics (1981) "Student use of computers in schools." News release, March, 20.

Osborne, Thomas (1977) "The personal computer and education," in Computers and Communication, Robert J. Seidel and Martin Rubin, eds.. N.Y.: Academic Press.

Papert, Seymour (1980) "New cultures from new technologies," Byte Magazine pp. 230-240 (September).

_____ (1981) Mindstorms: Children, Computing and Powerful Ideas. N.Y.: Basic Books.

Piele, Donald T. (1979) "Microcomputers 'GOTO' school." Paper presented at meetings of the American Educational Research Association, San Francisco (April).

Ragosta, Marjorie; Holland, Paul; and Jamison, Dean (1981) "Computer-assisted instruction and compensatory education: The ETS/LAUSD study." Final Report, Educational Testing Service.

Seltzer, Robert A. (1971) "Computer-assisted instruction--what it can and cannot do," American Psychologist 26: 373-377 (April).

Sheingold, Karen (1981) "Issues related to the implementation of computer technology in schools: a cross sectional study." Paper presented at the National Institute of Education Conference on Issues Related to the Implementation of Computer Technology in Schools, Washington, D.C. (February).

Shneiderman, Ben (1977) "Teaching programming: a spiral approach to syntax and semantics," Computers and Education. 1: 193-197.

Statz, J. (1973) "Problem solving and LOGO," in Syracuse University LOGO Report, Final Report, Syracuse University.

Tait, K.: Hartley, J.P.; and Anderson, R.C. (1973) "Feedback procedures in cc. iter-assisted instruction," British Journal of Educational Ps ·ology. 43: 161-171.

Uttal, W..liam R.; Rogers, Miriam; Hieronymous, Ramelle; and Pasich, Timothy (1969) "Generative C.A.I. in analytic geometry." Ann Arbor, MI: University of Michigan (November).

Vinsonhaler, J. and Bass, R. (1972) "A summary of ten major studies of C.A.I. drill and practice," Educational Technology 12: 29-32.

ERIC

Watt, Daniel (1981) "Teaching child-appropriate computing." Paper presented at Conference on Child-Appropriate Computing, Teachers College, Columbia University, (April 30-May 1).

Yaski, Edward (1981) "Kids love the D.P. Scene." <u>Datamation</u>.

ERIC