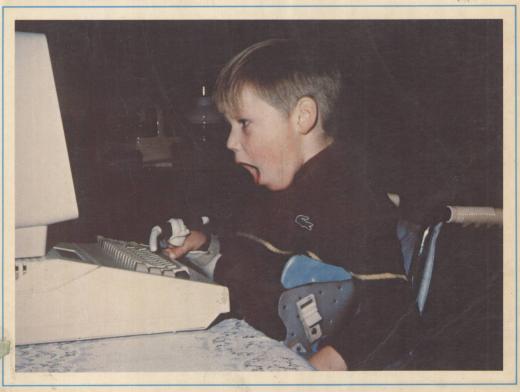# MICROCOMPUTER RESOURCE BOOK FOR SPECIAL EDUCATION

DOLORES HAGEN

Microcomputer Resource Book for Special Education

Dolores Hagen

Reviewed and commended to the field by
The Council for Exceptional Children

10   9   8   7   6   5   4   3   2   1

Microcomputer Resource Book
for
Special Education

To Marc
and all children who need technology for a better life

Trademarks Used in Microcomputer Resource Book for Special Education

Apple is a registered trademark of Apple Computer, Inc.

Apple II is a registered trademark of Apple Computer, Inc.

Apple IIe is a registered trademark of Apple Computer, Inc.

Apple II Plus is a registered trademark of Apple Computer, Inc.

Apple III is a registered trademark of Apple Computer, Inc.

Apple Cat II is a registered trademark of Novation, Inc.

Atari is a registered trademark of Atari, Inc.

CBM is a registered trademark of Commodore Business Machines

CCD is a registered trademark of Hartley Courseware, Inc.

Commodore is a registered trademark of Commodore Business Machines

CP/M is a registered trademark of Digital Research Corp.

DEC is a registered trademark of Digital Equipment Corp.

Diablo is a registered trademark of Xerox Corp.

Echo II is a registered trademark of Street Electronics

H-89 is a registered trademark of Zenith Radio Corp.

Hartley Courseware is a registered trademark of Hartley Courseware, Inc.

Heath is a registered trademark of Zenith Radio Corp.

IBM is a registered trademark of International Business Machines

PET is a registered trademark of Commodore Business Machines

TI99/4A is a registered trademark of Texas Instruments

Timex/Sinclair 1000 is a registered trademark of Sinclair Research, Ltd.

TRS-80 Model I, II, III are registered trademarks of Tandy Corp.

VIC-20 is a registered trademark of Commodore Business Machines

VisiCalc is a registered trademark of VisiCorp

Vox Box is a registered trademark of Tandy Corp.

Zenith is a registered trademark of Zenith Radio Corp.

64 is a registered trademark of Commodore Business Machines

800 is a registered trademark of Atari, Inc.

Introduction

Microcomputers are tools. They are a means to an end, just like any other tool devised by man. How effective the tool is, in the hands of the user, depends on the training and understanding that person has of the tool.

Today, a vast majority of the formal training to use a microcomputer is identified as some form of "computer literacy." One of the problems with this term is trying to find two people who agree on its meaning. In the view of some, it means an understanding of, and programming in, several computer languages. To others it means BASIC programming with a large portion of "computer history" thrown in. In very rare instances it can mean being a computer user.

It makes for interesting conjecture to think what rush-hour traffic in Los Angeles would be like if all the instruction for drivers' training classes had been "History of the American Auto Industry" or "How to Build a Carburetor" instead of user skills. Unfortunately, the computer training of today continues to stress programming and history instead of how to *use* the microcomputer.

The emphasis on learning programming and history of computers as an instructional set stems from the time when little or no software existed for microcomputers. If the micro was to be used, the individual had to develop his or her own programs since none were available for purchase. This most often meant learning to program in BASIC

(Beginners All-Purpose Symbolic Instruction Code) or even more complicated programming languages such as FORTRAN, COBOL and so on. Today it's a different story. There is a wide variety of commercially available software to choose from. "Basement" software developers and major publishers have provided a wide variety of educational and special applications software. True, it takes a little effort to find some of it (and not all of it is good), but there are reliable publications that review software both from a purpose and quality standpoint. Between print periodicals and computer data bases, access to current software information should be no problem. If someone wants to create their own courseware, there are skeleton programs called "authoring systems" ready to accept lesson materials just by learning a few simple directions and commands instead of complicated programming languages.

To be a microcomputer user is to use the machine and existing software to meet specific needs. From a technical point of view this means inserting the software, turning on the computer and following the directions given. To use peripheral devices you will have to learn to plug a few cables here and there, maybe even plug in a chip or two, but directions accompany these user modifications, or the device will tell you to "see your dealer" if the procedure is at all complicated. Users can now take full advantage of the computer software combination to complete a given task. It makes little difference what task the software is designed to achieve because the microcomputer will perform any task the software tells it to do. This makes the microcomputer a tool with thousands of uses, under direct control of the user.

The user concept allows anyone to reap the benefits of the computer. Age is no barrier nor is physical ability. There may be a problem of access for the physically handicapped but once the hardware and/or software interfaces have been established, the microcomputer is there for them as well. With today's technology, there is no reason computer access should not be available for everyone.

The material in this book covers the use of microcomputers in special education and the multitude of uses this technology offers the handicapped. All of what appears on these pages refers to user-oriented applications of the computer and requires no programming knowledge by the user. The information has been gathered as the result of the author being the parent of a handicapped child, a school district courseware developer, the publisher of a newspaper dealing with microcomputer usage by the handicapped, a computer workshop director and a computer *user*. I have not found it necessary to become a computer programmer nor do I expect to. The hardware and software exist to deliver educational and vocational benefits to the handicapped, just as they exist for all other people.

8566118

Contents

Overview

I

Microcomputers can provide a solution to many problems facing the handicapped. Today, a microcomputer can bring speech to the nonvocal, telephone use to the deaf, grade II Braille or voice to the blind and environmental control to the physically handicapped. It can remove the paper and pencil blockade for the learning disabled and improve the quality of life for the mentally handicapped. All of these dramatic uses of the microcomputer can be accomplished without knowing one word of programming.

These and thousands of other uses of the microcomputer can be implemented just "using" the machine. The delivery of this information, to the people who can, is the focus of this book.

There is so much to offer through the implementation of the microcomputer; however, one barrier exists that must be overcome before universal usage by the handicapped can be a reality. This barrier is not cost, or lack of technology. It is an informational one.

Within my own small circle of friends and relatives, myths and misinformation still prevail, such as:

The lady who says, "I can't use the microcomputer! I'm not a math major."

The teacher who stands by watching the microcomputer/printer and says, "I'll never be able to write a program to do that!"

1

The man who refuses to even sit down at the micro because, as he puts it, "I'll erase everything the computer is doing if I push the wrong key."

The parent who insists, "This whole computer thing is way over my head. I'll never be able to understand how the kids do it."

A similar list of comments can be heard by anyone who bothers to survey a school or neighborhood. If the person is over twenty-five and has never used a computer, there is likely to be something mysterious and intimidating about these machines.

Through no great fault of its own, the public has been buffaloed by the intimidating aspects of large computers. There is the massive computer control room of NASA as viewed on television just before a space launch, or the image of giant mainframes spitting out credit card billings, all very impersonal and complicated. The prevailing view of such matters is "genius at work," portraying an environment foreign to mere mortals.

Along came the microcomputer, also called the *personal computer,* and the public reacted to the "computer" part with awe and bewilderment. Even though this new era of computing used a different language, BASIC (touted as a "user friendly language"), the public was unwilling or unable to come to terms with the idea of learning a new language just to communicate with a machine.

Hobbyists began to program and pursue their goals, in GOSUB and IF/THEN statements. They could produce the software to make the computer do what they wanted, but what of the general population? They stood by watching and wondering. The computer was everywhere but still about as impersonal and unfamiliar as the old style mainframes. When the only game in town was the huge mainframe, not being able to program was expected. Suddenly, the computer industry was telling the world "anyone can do this, . . . in just a few hours, you too can be a programmer."

Nothing could have been further from the truth and nothing more damaging to the actual use of the microcomputer as a tool. The rift continued to grow between those who could use the machine and those who could not. The effect of that rift made those unable to program uneasy, even downright afraid of using the software the programmers developed.

There were those in education who recognized the long-term value of the microcomputer in their field. However, most of them assumed that programming was the only way humans could gain access to the micro. It followed that everyone should learn to program.

The microcomputer industry flourished and a new industry emerged. Software, tons of it, was being created by a cottage industry

of programmers. Soon major academic publishers joined the boom and *bang*, quality software to do just about anything was on the market.

Back in education, however, the momentum for teaching everyone to program was gathering. Buzz words like "computer literacy" and "CAI" (Computer Assisted Instruction) took over. The wheels were in motion. Educators braved the language barrier and began two major areas of computer usage in education. First, and still dominant, is the "Let's make everyone computer literate," a combination of history and/or programming, erroneously taught as the access to microcomputer usage. Second, there was the transformation of an old medium into the new box: take text from books and move it to the computer screen. Both are useful, but neither addresses the actual use of the microcomputer as a machine of varied uses to solve specific problems and neither explores the wealth of software that already has been written to allow the use of the microcomputer as a tool.

The microcomputer and existing software have the potential to make common computer usage a reality. Until the programming myths are laid to rest, however, the likelihood of effective implementation of this tool in education and for the handicapped is not very good.

In the early 1980s, the major obstacle was still software. Education's preoccupation with teaching programming resulted in a lack of awareness, much less use, of existent applicable software. This lack of awareness resulted in a total void of any comprehensive evaluation of software. There was no place to look except in the commercially produced software directories, wading through hundreds of listings to find one, maybe two, good programs in a sea of mediocre offerings. Part of this problem was attributable to the computer hobbyist attempting to create educational software but not knowing much about educational standards. The other was teachers, knowing a lot about education, but darn little about programming. The result—many useless hours of wheel spinning by both and a good many inappropriate programs sold as "microcomputer applications for education." It was hard to come away from software evaluation sessions without a sour taste in the mouth. Many educational computer pioneers were driven to near despair.

For those looking for special education applications, the situation was even worse. Nowhere was there even a partial listing of special-applications software. If you wanted to find something to serve the needs of the handicapped, you were at Ground Zero and nothing but a systematic search of all potentially useful software would solve the problem.

The software industry was growing with great speed, yet few educators were looking at what that industry was churning out to see if

it was usable. Those in a position to evaluate the programs were often too busy trying to produce their own software.

It wasn't that software didn't exist. In fact, it might be said that too much existed, with no way to distinguish the good from the bad, short of running every program.

It didn't take long before attempts were made to start weeding out the collection. Magazines directed to educators began reviewing software. For those interested, it was possible to get some idea of what certain programs were like before buying. However, the vast majority of programs were simply listed in an ever growing number of "software directories" with mere existence the only criterion for listing.

Adding to the difficulty of finding appropriate special-education programs was the nonlisted software. This was most often the case with special applications programs. Viewed as having limited appeal, these programs were very often not submitted to the directory publishers because their producers were solving an individual's problem and once that problem was solved, the software went on the shelf, never reaching production or marketing to the public.

With all this confusion about software, it is understandable why so few people were willing to pursue microcomputer use applications for the handicapped. Discovering appropriate educational software was a monumental task in itself. Locating special-education applications took Herculean effort and dedication.

Today, some of these problems still remain, but the situation has gotten much better. More and better software review systems are in place. Many periodicals exist that regularly review software in great detail. On-line data base systems that specialize in evaluation of educational software now are available. However, their contribution to the dissemination of information is limited to those with the special equipment that allows computer-to-computer access over telephone lines.

Another development that has improved software information is that major publishers and distributors have come on the scene. Through their efforts, large collections of high-quality software are much easier to identify than before. Many of these companies have begun their own software search and development divisions and the result is a continually growing library of quality software to choose from. "Where to find software" is largely a question of the past. However, it still leaves the question of "what software is applicable" up to the individual buyer.

There still are problems within the software and educational communities that make implementation of the microcomputer in special education difficult. These problems center around software

information getting to the people who can use it and identifying what software is actually appropriate for use. The first of those problems will not be solved until there is more organized access to pertinent information about handicapped usage and more people become familiar with microcomputer applications. The second, however, can be solved by following simple rules that address the software availability that exists and putting that software to work. Software availability is expanding daily. In a field with such rapid expansion, what is unavailable today may exist tenfold tomorrow. But once the general rules are understood and put to use, the new additions will simply enhance the existent usage.

General Rules To Finding Software

Software, that elusive part of the equation, is there—it's just a matter of finding it. As has been pointed out, there is little software listed as "special education software." This does not mean that none exists. It just means that the software industry has not addressed or understood the special education uses for its products. To solve this problem, identify the "need" and look for software to fill that need. If the child is twelve years old and in the sixth grade but having problems with math facts at the third grade level, there are hundreds of useful tutorial math programs that could be used. Most educational software is listed by grade level, thereby making the identification of special education applications a simple matter once the need has been determined.

The objective of "finding software" is best obtained by following Rule 1.

RULE 1   Software does not have to say "handicapped" on the label to be useful in special education.

The special education uses of the microcomputer are endless, especially if you take the labels off the software and apply what the programs do for the needs of the child. Take, for example, typing tutors. Traditionally typing is taught in the high schools. It has been thought of as training for those students headed for business careers. A far more realistic approach is to introduce typing skills in the elementary grades, via a typing tutor and a microcomputer. The reasons are obvious. First, these children will be expected to interact with microcomputers throughout their lives. The most efficient method of interaction now available is through the typewriter-style keyboard. Training for these children should include typing, the most

effective method of access to the very tool that can help them. This matching of software to need, though akin to Rule 1, develops into Rule 2.

RULE 2   Train handicapped children to use the microcomputer as efficiently as possible, as early as possible.

A teacher of learning disabled students put it very well when he said, "There is a lot of tedium in this business." Over and over again, repeat, repeat and repeat, again and again. For most learning disabled children, regardless of the disability, learning must include a very tedious repetition of information. This tedium, for both teacher and child, can stifle motivation. It can produce angry emotional outbursts from student and teacher alike. One-on-one tutoring of facts, drill and practice, repeating over and over again, is ideally suited for the microcomputer. First, it is an unemotional device that operates at the speed of the student. Second, the microcomputer is nonjudgmental except in its ability to choose correct or incorrect responses, thereby allowing the student hundreds of attempts without embarrassment or stigma attached to the repetition process. Third, it frees the teacher for more creative activities during one-on-one human encounters with the students. Fourth, children are far more motivated when the drill and practice is done on a microcomputer. As one teacher put it, "We couldn't get them to do these exercises at all, willingly. Now, we can't tear them away when the same material is on the microcomputer." This leads to Rule 3.

RULE 3   The microcomputer can provide motivating drill-and-practice for handicapped children.

Individual educational needs of the handicapped child have been recognized and served for a very long time. Each child is different and each has needs that are difficult and time consuming to serve. One-on-one teacher time, dedicated to serving those needs, has been the only way to bring the necessary help to these children—that is, until the advent of the microcomputer and programs called authoring systems. With authoring systems, it is possible to create specific courseware (learning materials used on microcomputers) tailored exactly to the need of each child, and you don't have to be a programmer to do it!

Authoring systems are tools. They are programs or procedural languages written as a skeleton to which instructional materials must be added. This skeleton is "bug free" and simply requires the addition of the desired lesson material to turn it into individualized courseware. The courseware prepared by authoring systems can offer a cost

effective alternative to the one-on-one teacher tutorial. Both drill-and-practice and concept reinforcement can be presented by microcomputer, not once during class time, but again and again, at home as well as in the educational environment. For the first time a tutorial package of motivating, stimulating materials can go with the child, to his or her environment, for as long as is necessary. For these and many other reasons discussed in later chapters, authoring systems are the base for Rule 4.

RULE 4   Educational use of the microcomputer for handicapped children should include individual courseware prepared with authoring systems.

Rules 1 through 4 deal with the microcomputer as an extension of the teacher or parent in the areas of tutorial needs. In this role the computer becomes as close to a one-on-one human encounter as is possible, without it actually being another human being. But this is also an extension of the "right or wrong," "black or white" side of education. Granted, it is a necessary side and the microcomputer meets those needs very well. However, this same microcomputer can become a tool on another side of education which is equally, if not more important. I often refer to this as "the need to release the caged intellect of the handicapped child."

Thinking and problem solving skills, which will allow a child to reason for himself, are very often neglected in the education of the handicapped. This often is the case because there is limited time and most of that time must be spent attempting to close the learning gap created by the handicap. There is a tendency to stress the acquisition of facts, the "right and wrong" side, because these elements appear to be necessary for survival.

As a tool to develop self-expression and reasoning skills, the microcomputer can play perhaps its most important role in aiding the handicapped child. By putting the child in charge of the microcomputer with a language like Logo, there can be a role reversal that makes the child the teacher. There is great truth in the old adage, "We learn by doing," and with Logo, the doing is original, yet the child is working with facts to accomplish his or her inventive creation. The child deals with elements of reality and is in charge of their organization, allowing for the creativity within the child to develop. In problem solving, building an organizational structure, uninhibited by the concepts of right or wrong, the child can explore his or her own ability to reason without an imposed pattern or instructional set.

Much the same effect can be achieved, in theory at least, with any computer programming. However, computer languages have made it