DOCUMENT RESUME

ED 244 493 EC 162 795

AUTHOR	Hummel, William; Hahn, Robert
TITLE	A Model Program in Microcomputer Utilization with Handicapped Students. Final Performance Report.
INSTITUTION	Linden Community Schools, MI.
SPONS AGENCY	Special Education Programs (ED/OSERS), Washington, DC.
PUB DATE	26 Dec 82
GRANT	G008100281
NOTE	120p.
PUB TYPE	Reports - Research/Technical (143) -- Guides - Classroom Use - Guides (For Teachers) (052)
EDRS PRICE	MF01/PC05 Plus Postage.
DESCRIPTORS	*Computer Assisted Instruction; *Computer Software; Elementary Secondary Education; *Emotional Disturbances; *Learning Disabilities; *Mathematics Instruction; *Mental Retardation

ABSTRACT

The report presents findings and materials developed in an investigation of the value of microcomputers and specially designed software materials in mathematics instruction with 90 learning disabled, emotionally impaired, and mentally retarded students in elementary, middle school and high school classes. Ss were divided into three groups: experimental (Ss received computer literacy training and used specially designed software materials); usage (Ss had access to the same hardware but had no literacy training and used only commercially available software); and control. Pre- and post-test scores on the Key Mathematics Test taken by all students measured achievement in Total Mathematics; the test also provided a Computation Skills subtest score which was used in analyzing project results. Results included differences at middle, elementary, and high school levels. Findings suggested that in the middle school, microcomputers may be more effective than traditional materials alone. Data from elementary and high school classes did not demonstrate more significant effectiveness for microcomputers. Difficulties in research design are noted as factors affecting levels of significance are identified. Also included in the report are an inservice guide for presenting material on classroom use of computers, a description of 17 computer remedial math programs, a computer instructional manual, and a manual to teach students to use the PET computer. (CL)

A Model Program in Microcomputer Utilization with Handicapped Students

(U.S. Department of Education
Grant #G008100281)

FINAL PERFORMANCE REPORT

William Hummel
Robert Hahn
Linden Community Schools
Linden, Michigan 48451
December 26, 1982

This report is divided into four sections: (A) a brief review of the objectives and methodology of this research project; (B) a summary of findings, and (C) a brief discussion regarding the potential implications of our results; and (D) an Appendix section which includes copies of all materials produced in conjunction with this project.

Further questions regarding any of this information should be directed to the co-directors of the project, William Hummel and Robert Hahn, Linden Community Schools, Linden, Michigan 48451, telephone 313-735-7821.

TABLE OF CONTENTS

A. METHODOLOGY

1. Objectives

The primary purpose of this project was to investigate the potential value of microcomputers and specially-designed software materials in the mathematics instruction of learning-handicapped students.

Specifically, we established these objectives:

(1) to determine whether the use of microcomputers with learning handicapped youngsters increased their rate of learning in mathematics;

(2) to determine if a comprehensive program using computer literacy training for teachers and students and specially-designed software resulted in greater gains in mathematics when compared with comparable classrooms using similar equipment and commercially-produced mathematics software, but without literacy instruction and specially-designed programming;

(3) to design a sequential set of basic instructional mathematics programs for use with learning-handicapped youngsters; and

(4) to provide these instructional programs to other interested school districts within Genesee County, Michigan and to develop a means for duplication and distribution throughout the state.

2. Methodology

This study involved approximately 90 students who had been formally identified as being learning disabled, emotionally impaired, or mentally impaired. These students were assigned, for a portion of the school day, to one of 21 special education classrooms within the Linden, Lake Fenton or Fenton school districts.

The 21 special education classrooms were divided into 3 groups, following the purpose of this study:

(1) an Experimental group, which received computer literacy training for students and teachers and used specially-designed software materials;

(2) a Usage group, which had access to the same amount of microcomputer hardware as the Experimental group, but did not undergo literacy training and used commercially-available software materials, and

(3) a Control group, which did not have access to hardware or software materials.

A sequential set of basic instructional programs in computational skills was written, refined, and duplicated for use in all Experimental group classrooms.

All students involved in this study were pre-tested and post-tested in Mathematic achievement, using the Key Mathematics Test. This test scores student achievement in Total Mathematics, and also provides a subtest score specifically in Computation Skills, which was used in analyzing the project results.

All classrooms in the Experimental and Usage groups were provided with identical inventories of microcomputer hardware. Whereas the Experimental group classrooms utilized specially-designed programs, the Usage group classrooms were provided with commercially-available Mathematics programs. Teachers in the Experimental group attended at least 10 hours of inservice on computer literacy.

B. RESULTS

For the purpose of this study we have accepted .10 as the accepted level of significance. This level was chosen because we view the study as a preliminary investigation into a relatively new area of research. Additionally, because of the field based nature of this study many important variables such as individual pupil time on the computer, uniform quality of computer time with respect to software, areas and degrees of student handicaps, as well as the investment with which individual educators approached the study, could not be controlled. The results may therefore be viewed as an indication of areas into which further research may prove productive.

There were 3 basic hypotheses which we investigated:

First, classrooms using computers would be more effective than traditional classrooms in the remediation of math deficits in learning handicapped students.

Second, computer using classrooms supported by a program of instructional and software support would be more effective than non-supported computer using classrooms in the remediation of math deficits in learning handicapped students.

Third, classrooms using computers would be more effective than traditional classrooms in enhancing student attitudes towards math.

A two way analysis of variance was performed on the three grade levels involved in the study (elementary, middle, and high school), versus the three Experimental groups (control, computer usage, and computer experimental). The analysis was performed with respect to total math gains for both full time students (students receiving all math instruction in the special education program), and full and part time students (students receiving all or part of their math instruction in the special education program). Similar analyses of variances were performed with respect to gains in specific computational math skills. No significant differences were found between the three conditions when all grade levels of students were included for either math gains or computational gains. However, in each case the differences between the 3 grade levels (elementary, middle and high school) were found to be significant at the .05 level (See Tables I through III and Figures I through IV). Because of the significant differences between the groups, data from each of the grade levels was re-evaluated. Records of computer usage indicated a lack of appropriate computer usage within several of the high school classrooms. Computer logs indicated that within these Usage and Experimental classrooms computer time for students was not only limited but was also almost totally geared toward non-instructional game activity. For these reasons, further analysis of results will be restricted to elementary and middle school students.

A two way analysis of variance involving two grade levels, elementary and middle school, and the three experimental groups indicated a significant difference in the predicted direction at the middle and elementary level between the three groups for total math gains with full time students, (p = .06). In addition, significant differences were found between the grade levels for following conditions:

Total Math - All Students (p=.05); Total Math - Full Time Students (p=.013); and Computation - Full Time Students (p=.08). See Table IV.

Because of the significant differences between the two grade levels, analyses of variance were performed for each group at the middle school and elementary levels. Significant differences were found at the middle school level for Total Math - Full Time Students (p=.08) and Computation - Full Time Students (p=.10). See Table V.

In order to assess the overall effectiveness of computer usage, a two-way analysis of variance was performed for two grade levels, elementary and middle school, versus two conditions, control and computer classrooms (combining data from the Experimental and Usage groups). Significant differences were found between the two conditions for Full Time - Total Math students (p=.04) and Full Time Computation students (p=.05). See Table VI. Further univariate analyses of variance were performed at the elementary and middle school levels for the two groups, control versus computer classrooms. Computer usage produced significant differences in the predicted direction at the middle school for all groups: All Students - Total Math (p=.09); All Students - Computation (p=.05); Full Time - Total Math (p=.04); and Full Time - Computation (p=.03). See Table VII.

A two-way analysis of variance was performed involving the three grade levels and three groups to measure any significant changes in attitude towards math instruction. No significant changes were present.

Analysis of usage logs revealed no significant data. There was a great deal of incomplete data at some grade levels, and statistical analysis of the logs was determined to be inappropriate.

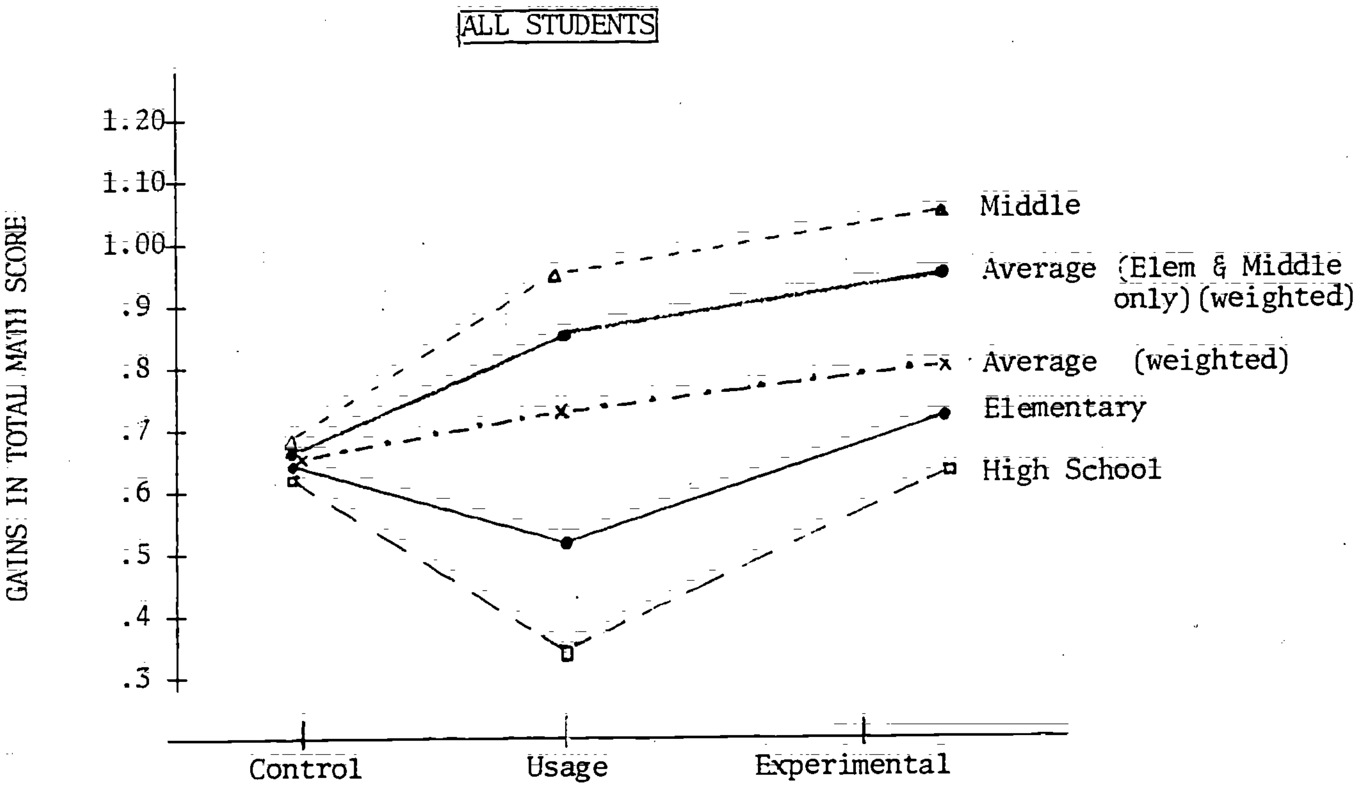


FIGURE I - Comparison of student gains in total mathematics across three experimental groups.

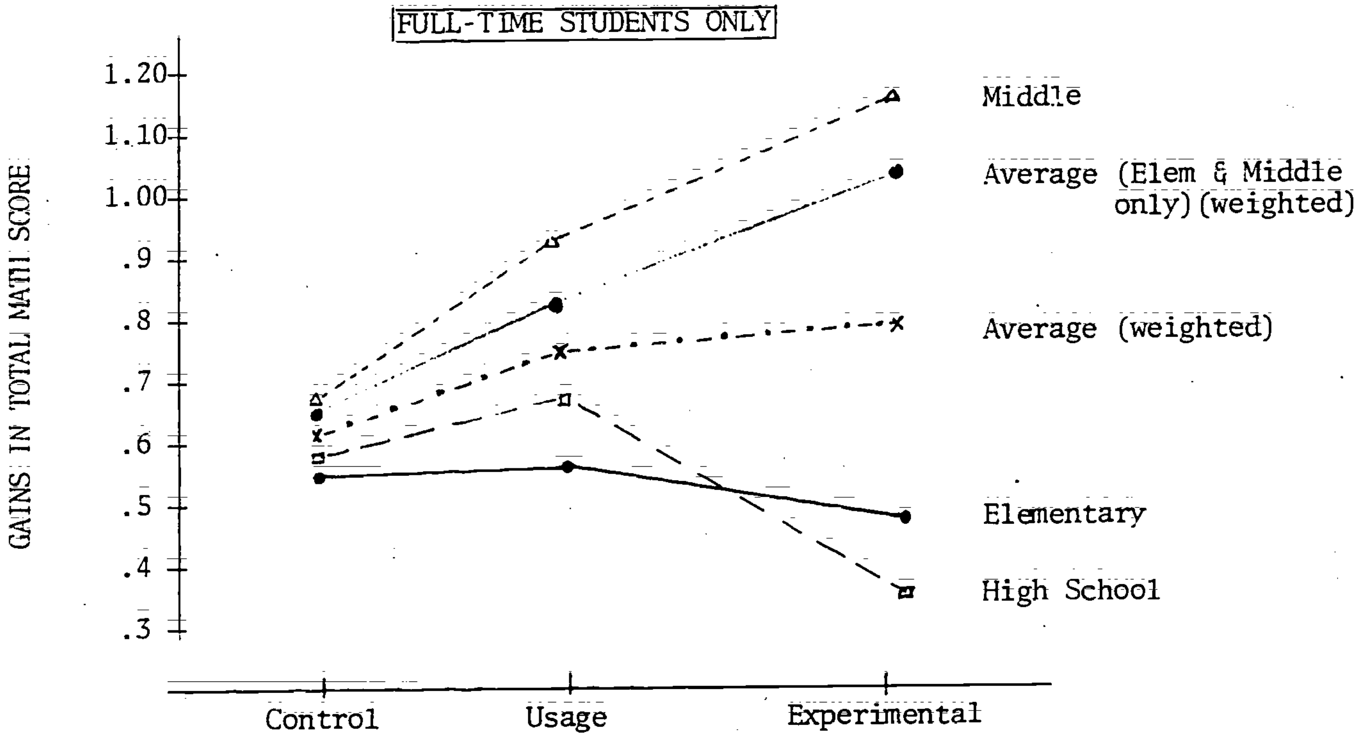


FIGURE II - Comparison of gains by full-time students in total mathematics across three experimental groups.

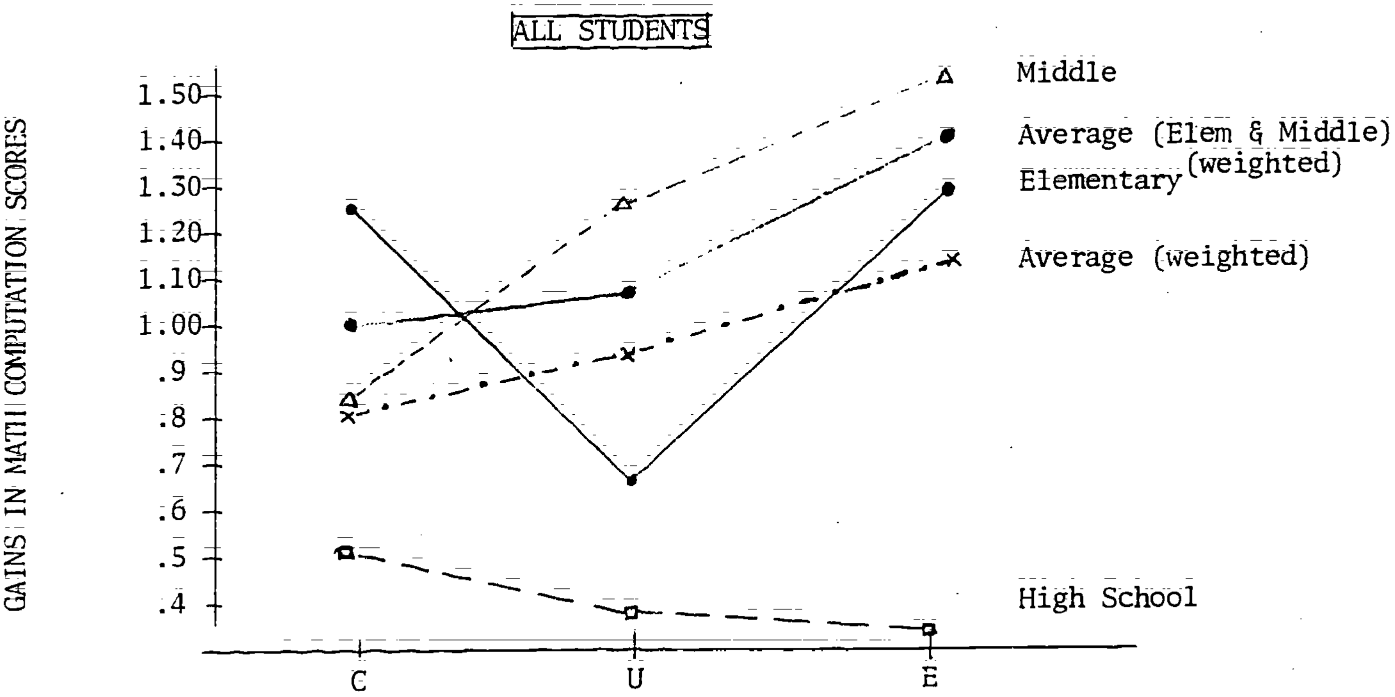


FIGURE III - Comparison of student gains in computation skills across three experimental groups.

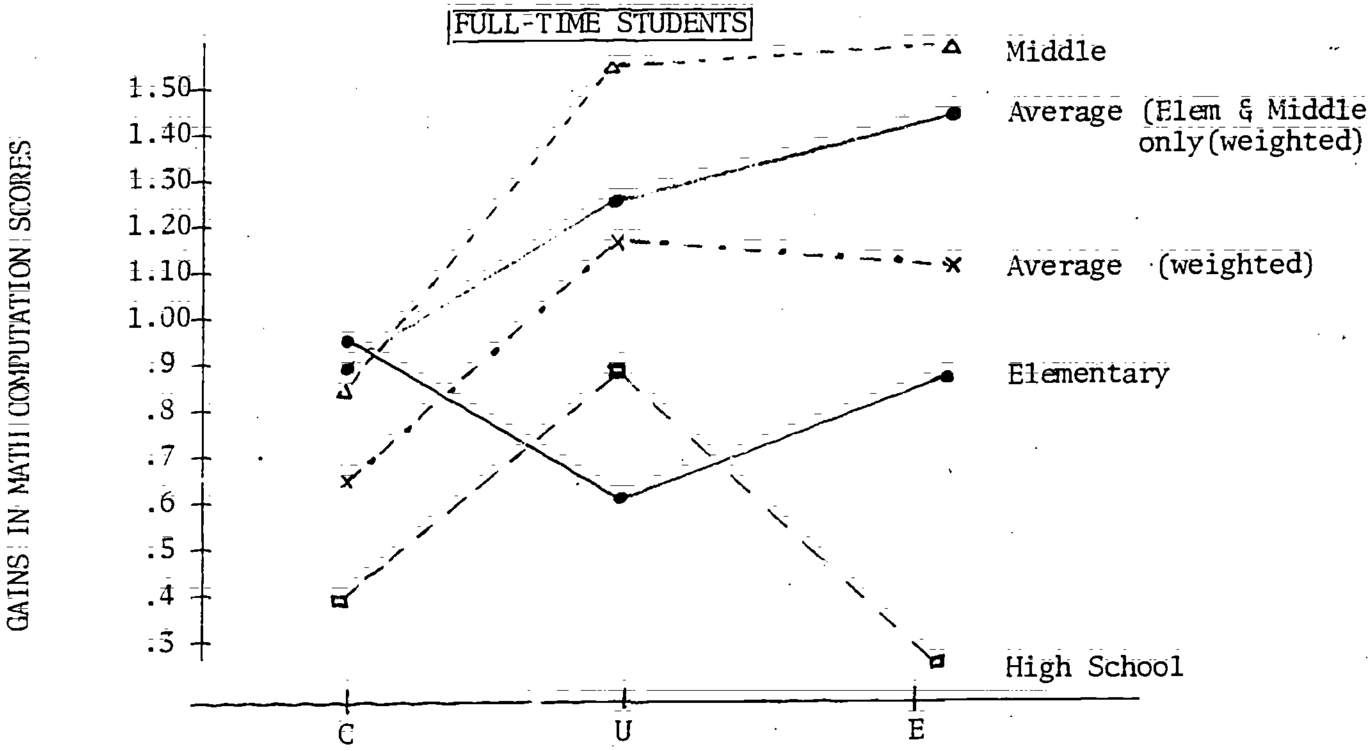


FIGURE IV - Comparison of gains by full-time students in computational skills across three experimental groups.

TABLE I

Two-Way Analysis of Variance
3 Grade Levels versus 3 Experimental Groups*

	F	df	p. Value
Total Math / All Students			
Control vs Usage vs Experimental	.96	(2,122)	.39
Elementary vs Middle vs High	3.71	(2,122	.03
Full-Time Students			
Control vs Usage vs Experimental	.83	(2,75)	.44
Elementary vs Middle vs High	5.12	(2,75)	.008
Computation / All Students			
Control vs Usage vs Experimental	1.67	(2,122)	.19
Elementary vs Middle vs High	13.01	(2,122)	.001
Full-Time Students			
Control vs Usage vs Experimental	2.01	(2,75)	.14
Elementary vs Middle vs High	8.96	(2,75)	.001

*Note-interactions were not significant for any group

TABLE II

Summary of Scores (Full-Time Students)

			Total Math		Computation	
		N	Mean	S.D.	Mean	S.D.
Control	- Elementary	6	.57	.38	.95	1.02
	Middle	12	.68	.56	.83	.81
	High School	13	.58	.90	.40	.63
Usage	- Elementary	4	.58	.46	.60	.48
	Middle	8	.91	.41	1.54	.91
	High School	4	.68	.48	.83	.33
Experimental	-Elementary	4	.48	.51	.88	.57
	Middle	21	1.16	.61	1.59	1.10
	High School	12	.35	.83	.26	1.22

TABLE III

Summary of Scores (All Students - Full & Part Time)

			Total Math		Computation	
		N	Mean	S.D.	Mean	S.D.
Control	-Elementary	9	.63	.37	1.24	.93
	Middle	13	.68	.54	.85	.78
	High School	14	.62	.88	.51	.74
Usage	-Elementary	5	.54	.40	.68	.45
	Middle	13	.95	.41	1.24	.86
	High School	6	.32	.71	.37	.76
Experimental	- Elementary	25	.76	.42	1.30	.71
	Middle	27	1.01	.62	1.53	1.03
	High School	19	.61	.84	.33	1.05

TABLE IV

Two-Way Analysis of Variance

2 Grade Levels versus 3 Experimental Groups*

Total Math - All Students	F	df	p value
Control vs Usage vs Experimental	1.62	(2,86)	.20
Elementary vs Middle	3.97	(1,86)	.05
Full Time			
Control vs Usage vs Experimental	3.04	(2,49)	.06
Elementary vs Middle	6.58	(1,49)	.013
Computation - All Students			
Control vs Usage vs Experimental	2.18	(2,86)	.12
Elementary vs Middle	.20	(1,86)	.66
Full-Time			
Control vs Usage vs Experimental	1.94	(2,49)	.13
Elementary vs Middle	3.21	(1,49)	.08

*Note - interactions were not significant for any group

TABLE V

e Analysis of Variance at Elementary and Middle School
Levels for Control vs Usage vs Experimental Groups

		f	df	p.value
al Math -	All Students	.82	(2,36)	.55
	Full-Time	.07	(2,11)	.94
putation-	All Students	1.47	(2,36)	.24
	Full-Time	.24	(2,11)	.79
al Math -	All Students	1.54	(2,50)	.23
	Full-Time	2.75	(2,38)	.08
putation-	All Students	2.33	(2,50)	.11
	Full-Time	2.43	(2,38)	.10

TABLE VI

Way Analysis of Variance (Elementary and Middle)
Control vs Computer Groups

	f	df	p.value
All Students	3.10	(1,88)	.08
Full-Time	4.42	(1,51)	.04
All Students	2.31	(1,88)	.13
Full-Time	3.87	(1,51)	.05

TABLE VII

Way Analysis of Variance (Elementary and Middle)
Control vs Computer Groups

		f	df	p.value
al Math -	All Students	.36	(1,37)	.56
	Full-Time	.03	(1,12)	.85
putation-	All Students	.03	(1,37)	.86
	Full-Time	.26	(1,12)	.62
al Math -	All Students	3.00	(1,51)	.09
	Full-Time	4.40	(1,39)	.04
putation-	All Students	3.85	(1,51)	.05
	Full-Time	4.98	(1,39)	.03

C. DISCUSSION

The results from this study are mixed.

Results from the middle school level indicate that microcomputers may be more effective than traditional materials alone at the middle school level; and that supportive programs for teachers and students when coupled with appropriate remedial software may enhance computer effectiveness. Not all evidence supports these conclusions. Results from the high school and elementary school levels did not show microcomputers to be significantly more effective. There may be several factors which may have contributed to the mixed results. As was mentioned in the results section, records of computer usage at the high school indicated that computer time for students was limited and not geared toward remedial instruction. Further, members of the high school staff asked not to participate in the study. This makes the high school data suspect as to its relevance to computer effectiveness when computers are used as an instructional tool. It does, however, raise the question of computer acceptance and appropriate usage among educators.

At the elementary level a different set of problems arose. First, due to the make up of the elementary programs, few students were placed for full time math remediation. Most students received basic math instruction in regular classrooms with special education support. The small number of pupils participating the Full-Time elementary group (n's of 6,4,4) makes these results of limited significance. Additionally, elementary educators at the lower grade levels have raised the question of appropriateness for microcomputer usage with young handicapped students. Some younger students with learning handicaps may not be able to successfully interact with microcomputers which involve keyboard entry of input, making microcomputers ineffective with these students. If computers can be made effective with these students other forms of software and hardware allowing easy manipulation of the material must be devised.

Another important variable between the groups may have been the degree of the instructors' experience with microcomputers. At the middle school level all computer using teachers had had prior experience with microcomputers in the classroom, whereas at the elementary level most computer using educators had had no prior computer experience. It may be that because of the lack of experience educators have with this technology an opportunity to use this technology with students is necessary before optimal use can be made in the classroom.

Upon analysis of the results at the middle school level it appears that the most significant results were obtained with the Full Time students, students receiving all of their math instruction within the special education classroom. It may have been that these students had the computer utilization more closely integrated into their math curriculum thus providing optimal benefits for the remediation of their deficits. Further study of this area may be warranted.

Another factor affecting levels of significance was the large scatter of scores within each group. In each group, some students made large gains in math while others made little gains and a few students even posted losses. This inconsistency in testing is common with learning handicapped students. To overcome the effect of this scattering much larger populations of students would be needed to determine significant measures.

In the same area some teachers noted different levels of acceptance of the computers. Some students appeared highly motivated for computer usage attempting to gain additional computer time while a few students attempted to avoid computer usage. With most handicapped students there does not appear to be one method or type of material which is effective for all students. It may be that computers are effective in remediating math deficits for some students and ineffective for others. Further study in identifying students who may benefit by computer usage may be useful.

Other factors which may have affected levels of significance were the variation in time spent on the computer by students as well as the programs used. The amount of time and types of programs used by individual students were left to the discretion of individual classroom teachers and not controlled with study. The amount of computer time should have been controlled for each student, as well as the nature of the the remedial programs (similar remedial programs were supplied to all classes but their use was not controlled). This variance in the quality and quantity of computer time for each student may have affected the levels of significance and should have been controlled in order to provide a strict measure of computer effectiveness.

APPENDIX

SPECIAL EDUCATION COMPUTER MANUAL

PRESENTOR'S MANUAL

developed under Federal Grant #G008100281

Model Program in Microcomputer Utilization with Handicapped Students

by Paulette M. Hummel, M.A.
Fenton Area Public Schools

and

William R. Hummel, M.A.
Linden Community Schools

INTRODUCTION

This manual was developed for use as an inservice guide for presentors, to instruct their colleagues in the classroom utilization of computers. It assumes that, as a presentor, he/she has knowledge of microcomputers and has had experience using them with students as an instructional tool. The topics outlined for each session are merely suggestions and can easily be altered to fit the audience and time constraints in various school districts. It is hoped that this manual will be helpful in introducing microcomputers in the schools.

Inservice Materials

The inservice package contains the following items:

1. Presentor's Manual
2. Participant's Manual
3. Student Manual
4. Math programs and accompanying manual

1. Presentor's Manual--This manual is a guide for someone who has used microcomputers with students and will be instructing teachers in their use.

2. Participant's Manual--This manual covers the operation of the PET computer, explanations of computer programs, a guide to available software, a checklist for evaluating software, a glossary and bibliography. The manual will be used as an instructional tool during inservice sessions, as well as a reference guide for the teacher in the classroom.

3. Student Manual--This is a manual that can be used as a lesson guide for a group or used as a workbook for individual student use. It instructs students in the usage and care cf the PET computer. An optional section is included for more in-depth study.

4. Math programs and accompanying manual--The math package consists of 14 programs covering the basic computational skills in addition, subtraction, multiplication and division. These programs provide problems to be worked on the screen as well as hard copies of results and problems for the user. Also included are 3 additional programs covering place value, rounding, and estimating. An infinite number of problems can be produced where the parameters can be controlled by the teacher. The manual explains each program in detail.

Purpose of Inservice

In order for computer utilization to be successful as an instructional tool teachers must first become familiar with the operation of the equipment. Secondly, they must be given various methods and strategies which can be implemented within classroom settings. Information regarding sources and types of software currently available, should be offered. Lastly, the opportunity for on-going dialogue must be provided to discuss the effectiveness of programs, needs which should be addressed and any problems which have arisen.

When familiarizing teachers with microcomputers, several terms need to be defined before beginning hands on experience. The most basic of these are included in the glossary in the participant's manual. Once the participants have a fair grasp of these terms they can begin to learn the keyboard and the functions of the keys. The sessions should progress to operating the microcomputer using simple commands: i.e., loading and saving programs. It is important to note that the teachers should read the manual to reinforce and supplement the inservice presentations. This will help to allow more time for hands on experience during the sessions.

Once the teachers feel competent in operating the microcomputers, discussions should be held on the variety of ways it can be used in the classroom setting. Teachers will also need to know how, where, and at what cost software can be obtained. Throughout the inservice sessions the teachers should be encouraged to ask questions and discuss any successes or problems they may have had or needs they would like to see addressed. This last item is highly important as all teachers have a unique teaching style and classroom setting with different needs. Thus the inservice presentor must attempt to meet a variety of needs. By creating an atmosphere conducive to an open dialogue of sharing ideas, this objective can be met.

Objectives and Strategies

Session 1

Before the first session begins the participants should be given their manuals. A brief amount of time should be allotted so that they can scan the contents.

The presentor should then begin the instruction of the keyboard, explaining the function of the keys. This is covered in pages 4-8 of the participant's manual. Time should be given for them to try the exercises included in that section. They should be encouraged to read pages 9-10 (using the PET as a calculator) on their own. As the PET operates very similarly to commercial calculators, inservice time is not necessary for that purpose.

The next topic to be covered, the loading and running of programs, deserves a large block of uninterrupted time for it is the core of information needed to utilize the computer in the classroom. As most classrooms use the cassette drive, this method of loading programs should be taught first. If any of the participants have a disk drive system, that can be taught after the cassette loading method. A distinction must also be made between the 3.0 and 4.0 Basic machines. However, the method for loading programs (via cassette) on the 4.0 machines can be used on the 3.0 machines as well.

Ample time should be given to ensure that each participant can load a program. Any programs can be used to teach this skill. All participants should be given the opportunity to run at least one program. (This information is covered on pages 11-17 in the participant's manual.)

Session 2 Objective: All participants will be able to copy a program via disk or cassette. All participants will be able to operate a printer.

When instructing the participants in copying programs, only public domain programs should be used. Using any other programs is a violation of copyright laws. The cassette method should be taught first, allowing each participant to copy one program. Although any brand of cassette tape may be used, better results are obtained if cassette tapes made for computers are used. (This instruction is covered on pages 12-14 of the participant's manual.) Disk copying may be taught next and is covered on pages 15-16 of the participant's manual. To avoid future loading errors it is wise to verify each program after it has been copied.

After all participants have copied a program, the printer can be introduced. Very little instruction is necessary other than the hook up and paper insertion. The printer is connected to the PET with a PET to IEEE cable. The cable can only be inserted one way so there is no chance for error. The proper paper insertion is crucial. If not inserted correctly, the printer will not print. The paper must be inserted behind the rocker and over the yellow metal clip on the left. Detailed drawings are included in the manual which accompanies the printer. The printer should always be turned on <u>after</u> the PET and turned off <u>before</u> the PET. The advance paper button, the red button on the right, will advance the paper 1 full sheet. It is not necessary to hold the button down while the paper is advancing.

During programs which utilize the printer a question will usually appear on the screen: "connect printer and press any key". All that is necessary is for the printer to be turned on, then any key on the PET can be used to activate it.

<u>Session 3</u> Objective: All participants will be able to clean and troubleshoot their machines. All participants will be able to edit a program.

To ensure the machines remain in good working order they should be properly maintained. When using a cassette drive it is important to keep it clean and free of dust. This will prevent loading errors and give better copying results. The only equipment needed is alcohol and cotton swabs. This process is described on page 18 of the participant's manual. It will take only a few minutes for the participants to clean their cassette. Page 23 of the participant's manual lists several common problems and their solutions. These should be discussed in depth. This will not only avoid 'down time' in the classroom but prevent unnecessary service calls as well. On page 22 of the participant's manual is a list of helpful hints. Again, these simple procedures can avoid maintenance problems and prolong the life of the hardware.

Editing programs is a simple operation, whereby teachers may tailor certain programs to meet individual student's needs. This instruction is covered on pages 19-21 of the participants manual. Many programs have the instructions printed in the directions or they can be found when the program is listed. Suggested programs to edit are: Flash, U-do-it-spell, Hangman, and Vocab-U-Pet. All participants should be given an opportunity to edit a program and save the results.

Session 4 Objective: All participants will be able to categorize a sampling of programs. All participants will be able to evaluate programs for use in their classroom.

As software is essential to the effectiveness of the computer in the classroom, it is important for the participants to know the different types available. A listing of these types and examples of programs are found on page 25 of the participant's manual After discussion of each type, one of the sample programs should be loaded for the participants to view. Then, other programs can be shown and the participants should categorize them by type. Since programs were designed for different purposes, each participant must decide if a particular program is suited for his/her students.

Session 5 Objective: All participants will design a management system for their classroom.

Now that the participants can operate the hardware it needs to be implemented into the classroom. This is discussed on pages 26-34 in the participant's manual. Each teacher must tailor the usage to fit his/her classroom needs.

This session can also be used for additional instruction or problem solving, if necessary. The presentor may also wish to point out the student's manual at this time. This is meant to be a guide. When students are instructed in computer usage and can operate it independently, it eliminates teacher time which can then be used for direct teaching of the students.

COMPUTATIONAL REMEDIAL MATHEMATICS

C*R*M

Developed Under U.S. Department of
Education Project #G008100281

Model Program in Microcomputer Utilization with Handicapped Students

William R. Hummel, M.A.
Linden Community Schools

All materials developed under this project are protected by copyright and are not to be reproduced for sale. However, copying and disemination of any and all materials is encouraged for educational usage with learning handicapped children provided no charges are made over and above the costs of materials and time involved.

The intent of the author is to allow interested users free access to any and all materials with no one profitting financially from their replication or distribution.

Materials are available for distribution throughout Michigan at Michigan Regional Eduaction Materials Centers (REMCS).

Index

Rationale

The Computational Remedial Mathematics (C*R*M) programs are a set of 17 computer programs designed for use with the Commodore PET. These programs provide drill and practice in addition, subtraction, multiplication, and division; as well as, place value, rounding, and estimating. The programs allow the teacher to produce problems for practice in specific skill areas for remediation of computational deficits. Problems produced are interactive with the computer providing immediate feedback to the user. If the programs are run with a printer a permanent record of the student's results may be produced. Paper and pencil drill sheets may also be produced for problems in addition through division.

The C*R*M programs are not designed to be use as a complete mathematics curriculum. A sound mathematics curriculum is comprised not only of drill and practice with computational algorithms but also instruction in the use of the algorithms, as well as, practice with math applications and underlying concrete and abstract mathematics principles. C*R*M does, however, provide a necessary supplement to any elementary math curriculum. Some of C*R*M's benefits are:

Motivation. A highly motivating format for lowly motivated students.

Feedback. Immediate feedback for students who are unsure of their skills.

Specificity. Practice in specific and isolated skill areas.

Repetition. Provides an infinite number of different problems in any skill area for students who need a lot of practice to master a skill.

Variability. The difficulty levels within specific skill areas can be controlled allowing students to be introduced to an algorithm even though they may not have mastered all skills normally associated with that algorithm.

The format of the C*R*M programs allows the teacher to design the type of problems to be produced by answering a series of questions at the beginning of each program. This format relies on the teacher's competance within the mathematics curriculum to develope problems which meet the needs of individual students. The more knowledgeable the teacher is in math instruction the more effectively the programs can be used. However, with a little practice they can prove useful to any classroom teacher.

Operational Notes (Screen Interaction)

If the user selects problems to be presented on the screen problems will be presented one at a time. The computer works through each problem with the user just as if working the problem with paper and pencil. The computer prompts each response with a question mark '?'. Unless directed by the computer the user should not type complete answers, but should type only the number which fits the digit being calculated. For example, in a multiplication problem the screen might show:

```
  14
X 12
----
   ?
```

The correct response is 8. The screen would next show:

```
  14
X 12
----
  ?8
```

etc..

If the screen showed:

```
                     7
Type Complete      X 6
Answers            ---
                     ?
```

the correct

response would be 4 2.

In subtraction problems which may require borrowing the screen will print "Press 'B' to Borrow". When the user comes to a digit where borrowing is necessary the user must press 'B' and respond as if borrowing with paper and pencil. Example:

$$\begin{array}{r} 618 \\ -\ 27 \\ \hline ? \end{array} \qquad \begin{array}{r} 618 \\ -\ 27 \\ \hline ?1 \end{array} \qquad \begin{array}{r} ?\ \ \\ 618 \\ -\ 27 \\ \hline B1 \end{array} \qquad \begin{array}{r} 5\,11\ \ \\ \cancel{6}\cancel{1}8 \\ -\ 27 \\ \hline ?1 \end{array} \qquad \begin{array}{r} 5\,11\ \ \\ \cancel{6}\cancel{1}8 \\ -\ 27 \\ \hline ?91 \end{array} \qquad \begin{array}{r} 5\,11\ \ \\ \cancel{6}\cancel{1}8 \\ -\ 27 \\ \hline 591 \end{array}$$

In problems which may involve carrying the computer will print "Press 'C' to Carry". This operation is optional. If an operation involves carrying and the user presses 'C' then a question mark will be placed above the next digit and the amount to be carried must then be inputed before continuing. If the user does not wish to physically carry he may input the correct answer for the next digit without pressing 'C'.

The teacher should familiarize himself/herself with any program before presenting it to students for classroom use.

Addition 1:2

This program involves the addition of 2 numbers where the addends and augends are numbers between 0-9. The following options are available:

I. Sums of Ten

The addend and sum of 10 is presented. The user supplies the augend.

```
Example:    6
          + ?
           10
```

II. Mixed Sums

The addend and augend are presented. The user supplies the sum.

```
Example:    5        8
          + 4      + 9
            ?        ?
```

A. Addition of Families

The problems may contain either:

1) One family (0-9)

2) A choice of families (0-9)

3) All families (limited to paper and pencil sheets)

B. Addition of Random Problems

The range of digits may be limited to contain:

1) Addends 0-5

2) Addends 5-9

3) Addends 0-9

Addition 1:2 cont.

The addition of random problems may also be controlled for sums. Sums may be limited to contain:

1) All sums less than 9

2) Mixed sums (0-18)

Addition 1:3-5

This program involves the addition of a column of 3-5 numbers where each number is less than 10.

Example:

$$\begin{array}{r} 3 \\ 4 \\ +\ 7 \\ \hline \end{array} \qquad \begin{array}{r} 4 \\ 3 \\ 8 \\ +\ 2 \\ \hline \end{array} \qquad \begin{array}{r} 6 \\ 8 \\ 2 \\ 5 \\ +\ 1 \\ \hline \end{array}$$

The following options are available:

I. Regrouping of Tens

Depending on the number of digits chosen, either 1 or 2 pairs of numbers may be grouped to form sums of 10. The user may select 3, 4, or 5 digits.

Example:

$$\begin{array}{r} 6 \\ 3 \\ +\ 4 \\ \hline \end{array} \qquad \begin{array}{r} 7 \\ 2 \\ 3 \\ +\ 8 \\ \hline \end{array} \qquad \begin{array}{r} 5 \\ 1 \\ 9 \\ 3 \\ +\ 5 \\ \hline \end{array}$$

II. Controlled Addends

The range of digits within each problem may be limited to:

1) 0-5

2) 5-9

3) 0-9

Addition 2:2-5

This program involves 2 column addition ranging from 2-5 rows.

Example:

$$\begin{array}{rrrrr}
 & & & & 71 \\
 & & & 62 & 32 \\
 & & 47 & 3 & 5 \\
27 & 32 & 30 & 36 & 20 \\
\underline{+\ \ 5} & \underline{+\ 61} & \underline{+\ 18} & \underline{+\ 29} & \underline{+\ 41}
\end{array}$$

The following options are available:

I. Two Row Addition

A. 2 Digit + 1 Digit

1) no carrying

2) random carrying

3) carrying - one's place only

4) carrying in one's place, random carrying in ten's place

B. 2 Digit + 2 Digit

1) no carrying

2) random carrying

3) carrying in one's place only

4) carrying in ten's place only

5) carrying in one's place, random carrying in ten's place

6) carrying in ten's place, random carrying in one's place

7) carrying in the one's and ten's place

Addition 2:2-5 cont.

II. Multi-row Addition (3-5 rows)

Numbers within individual digits may be limited to:

1) 0-5

2) 5-9

3) 0-9

A. Addition of Multiples of 10

Example:

```
                 80
          60     70
   40     60     90
   50     20     90
 + 30   + 90   + 80
```

B. Mixed Numbers

1) Numbers ten or greater

Example

```
                 47
          99     22
   10     42     61
   34     63     56
 + 82   + 81   + 72
```

2) Numbers between 1 and 99

Example

```
                  1
          48     14
   92      6     90
    6      5      2
 + 71   + 79   + 88
```

Addition 3:2-5

This program involves 3 column addition ranging from 2-5 rows.

Example:

```
                                                319
                                      473       465
                            676       691       171
    735        926          492        22       727
  + 372      +  94        + 110     + 406     + 233
  -----      -----        -----     -----     -----
```

Numbers within individual digits may be limited to:

1) 0-5

2) 5-9

3) 0-9

The following options are available:

I. Addition of Multiples

Problems may involve either:

A. Addition of Tens

B. Addition of Hundreds

Examples:

```
     40       300
     50       600
     20       700
     70       600
   + 30     + 100
   ----     -----
```

II. Mixed Addition

Number sizes may range from:

A. Numbers from 100-999

B. Numbers from 10-999

C. Numbers from 0-999

Addition 4:2-5

This program involves 4 column addition ranging from 2-5 rows.

Example:

```
                                   4382
                         7821        78
             6893        3220      7010
   6941      7672        7682      2000
 + 7003    + 4128      + 8719    +  493
 ------    ------      ------    ------
```

Numbers within individual digits may be limited to either:

1) Numbers 0-5

2) Numbers 5-9

3) Numbers 0-9

The following options are available:

I. Numbers ranging from 1 to 999

Example:

```
                                   7890
                           98        45
             4567         789         9
     91         8          67      9336
 + 7898    +  611      + 8700    +  755
 ------    ------      ------    ------
```

II. Numbers ranging from 1000 to 9999

A. Addition of Multiples

1) Tens

2) Hundreds

3) Thousands

B. Addition of Mixed Numbers

Addition 2-4:2C

This program involves the addition of two numbers. The size of the numbers may be varied from 10-9999.

```
Example:     24        99       992       698       7904
           +  2      + 48     +   4     + 247     + 6992
```

The user selects the number of digits (2-4) to be used in the addend as well as the number of digits (1-4) to be used in the augend.

The following options are available:

I. Use of zeroes. Zeroes may be:

A. Prohibited

B. Random

C. Controlled in selected digits of both the addend and augend. Within each digit zeroes may be:

1) Prohibited

2) Random

3) Required

II. Carrying. Carrying may be:

A. Prohibited

B. Random

C. Controlled in selected digits. Carrying may be:

1) Prohibited

2) Random

3) Required

(Note: the ability to control carrying within each digit is dependent upon the selection of zeroes.)

Subtraction 1

This program involves the subtraction of 2 numbers where the subtrahend and minuend are numbers between 0-9.

Example: $\begin{array}{r} 9 \\ -\ 2 \\ \hline \end{array}$ $\quad\begin{array}{r} 7 \\ -\ 5 \\ \hline \end{array}$ $\quad\begin{array}{r} 8 \\ -\ 1 \\ \hline \end{array}$

The following options are available:

I. Random Subtraction

Any single digit numbers which produce a non-negative answer are possible.

II. Subtraction by family

Example: 9 family $\quad\begin{array}{r} 9 \\ -\ 4 \\ \hline \end{array}$ $\quad\begin{array}{r} 9 \\ -\ 6 \\ \hline \end{array}$ $\quad\begin{array}{r} 9 \\ -\ 2 \\ \hline \end{array}$

6 family $\quad\begin{array}{r} 6 \\ -\ 5 \\ \hline \end{array}$ $\quad\begin{array}{r} 6 \\ -\ 3 \\ \hline \end{array}$ $\quad\begin{array}{r} 6 \\ -\ 6 \\ \hline \end{array}$

Options:

A. All families 0-9

B. From 1-10 families

(Note: if problems are presented on the screen, only 2 families may be chosen.)

Subtraction 1-4

This program involves the subtraction of two numbers where the subtrahend and minuend are numbers between 1 and 9999.

```
Example:     16        93       681       681      3472
           -  5      - 42     -   5     - 527    - 1223
```

The following options are available:

I. Restricted Minuend (10-19 subtraction)

All numbers in the minuend may be limited to numbers 10-19 and the subtrahend limited to numbers 0-9.

```
Example:     12        19        13
           -  6      -  8      -  9
```

II. Standard Subtraction

A. Use of zeroes may be:

1. prohibited

2. random

3. controlled in selected digits of both the minuend and subtrahend. Zeroes may be prohibited, random, or required in each digit.

B. Use of borrowing in selected digits may be:

1. prohibited

2. random

3. required (Note: borrowing is dependent upon the use of zeroes within digits.)

Multiplication 1

program involves one digit times one digit multiplication.

iple: $\begin{array}{r} 8 \\ \times\ 6 \\ \hline \end{array}$ $\begin{array}{r} 5 \\ \times\ 9 \\ \hline \end{array}$ $\begin{array}{r} 7 \\ \times\ 7 \\ \hline \end{array}$

following options are available.

iplication of families

problems may contain either:

A. All families 0-9

B. 1 to 10 families
(Note: if problems are presented on the computer the user is limited to a maximum of two families.)

om Problems

numbers within the multiplicand and multiplier may be pendently controlled. Numbers may be limited to:

A. 0-5

B. 5-9

C. 0-9

Multiplication 1-4

This program involves the multiplication of numbers with from 2-4 digits in the multiplicand and 1-4 numbers in the multiplier.

```
Example:      42        45        897        4974
            X  8      X 63      X 972      X   16
```

The following options are provided:

I. Difficulty Levels

The difficulty of problems may be controlled by limiting the range of numbers used in individual digits. Numbers may range from:

A. 0-5

B. 5-9

C. 0-9

II. Use of Zeroes.. Zeroes may be:

A. Prohibited

B. Random

C. Controlled in selected digits of the multiplicand and multiplier. Zeroes may be:

1) Prohibited

2) Random

3) Required

(Note: digital guidelines are available on paper and pencil sheets for 2 and 3 digit problems.)

Division 1

This program involves problems with 1 digit divisors and 1 digit quotients and no remainders.

Example: 4 / 36 3 / 64 3 / 15

The following options are available:

I. Division by Families

The user may select sets of problems involving selected families.

Example: 6 / 30 6 / 12 6 / 36

II. Random Division

Problems are presented in random.

Division 1:2-4

This program involves problems with one digit divisors and 2-4 digits in the dividend.

Example: $8 \overline{)39}$ $6 \overline{)360}$ $7 \overline{)4826}$

The following options are available:

I. Digits Within the Quotient

The number of digits within the quotient may be controlled to be:

A. Equal to the number of digits in the dividend

B. One less than the number in the dividend

C. Random

II. Remainders. Remainders may be either:

A. Prohibited

B. Required

C. Random

III. Zeroes

Zeroes may be controlled in selected digits of either the dividend or quotient. (Note: zeroes may be controlled only if remainders are random.)

Zeroes may be either:

A. Prohibited

B. Required

C. Random

IV. Divisor

The divisor may be limited to numbers from 2-5 or 2-9

Division 2:2-5

This program involves problems with two digit divisors and 2-5 digit dividens.

Example:

$24 \,/\, \overline{86}$ $\quad$ $82 \,/\, \overline{943}$ $\quad$ $67 \,/\, \overline{4276}$

The following options are available:

I. Digits within the Quotient

The number of digits within the quotient may be controlled to:

A. Equal the number of digits in the dividend

B. Be one less than the number in the dividend

C. Be random

II. Remainders. Remainders may be either:

A. Prohibited

B. Required

C. Random

III. Zeroes (Dividend or Quotient)

(Note: zeroes may be controlled only if remainders are random.) Zeroes may be either:

A. Prohibited

B. Required

C. Random

IV. Divisors. Divisors may be limited to numbers from:

A. 10-99

1) All numbers 10,11,12,......99

2) Multiples of ten 10,20,30,......90

Division 2:2-5 cont.

B. 10-50

1) All numbers 10,11,12,......50

2) Multiples of ten 10,20,30,......50

Division 3:3-5

This program involves problems with three digit divisors and 3 to 5 digit dividends.

Example:

$481 \overline{)546}$ $610 \overline{)2349}$ $981 \overline{)69878}$

The following options are available:

I. Remainders. Remainders amy be either:

A. Prohibited

B. Required

C. Random

II. Zeroes (Dividend). Zeroes within individual digits of the dividend may be:

A. Prohibited

B. Required

C. Random

III. Zeroes (Divisor). All divisors may include zeroes in the:

A. One's digit

B. Ten's digit

C. One's and ten's digit

D. Random Usage

Supplemental Programs

Place Value

This program provides review and practice in the skill of identifying and naming digits from one's to million's.

Rounding

This program provides review of place value and rounding; and practice in the skill of rounding numbers to the nearest ten through ten thousand. Three different ranges of problems may be chosen.

Estimating

This program provides drill and practice in the skill of estimating answers to addition, subtraction, multiplication and division problems.

SPECIAL EDUCATION COMPUTER INSTRUCTIONAL MANUAL

developed under Federal Grant #G008100281

Model Program in Microcomputer Utilization with Handicapped Students

by Paulette M. Hummel, M.A.
Fenton Area Public Schools

and

William R. Hummel, M.A.
Linden Community Schools

TABLE OF CONTENTS

INTRODUCTION

The purpose of this manual is to acquaint the user with the Commodore PET and to aid in its use in the classroom. The topics covered are:

I) Computer Operation
II) Trouble Shooting
III) Software Applications
IV) Classroom Use and Management
V) Software Evaluation
VI) Software Guide
VII) Answer Key
VIII) Glossary
IX) Bibliography

Throughout the manual there are exercises for you to try. If you have difficulty with these, re-read the corresponding section. Answer keys are provided where appropriate.

The Commodore PET (subsequently referred to in this manual as the PET) is a relatively simple machine to operate. It is a self-contained computer which means it is capable of receiving, processing and storing information without the use of an outside source (such as a large computer often located in an intermediate school district center). Information may be stored via a cassette tape recorder or a disk drive, both of which are discussed later in this manual.

The PET was designed for school use and is a very sturdy machine. You can type virtually any combination of keys without damaging the machine.* Thus, do not be afraid if you accidentally hit two or more keys at once or press a wrong key.

* Actually there is one machine language POKE command which could damage the screen. However, the odds of hitting the command and the subsequent number sequence is extremely remote.

COMPUTER OPERATION

A. Turning On The Machine

The power switch is located on the bottom left side on the back of the machine. When turned on the screen will print:

```
###Commodore Basic###
15359 BYTES FREE
READY.
```

The number of bytes which appears determines the machine's memory size. A machine with a larger memory can store a larger amount of information and may be instructed to perform a longer series of operations.* Depending on the type of machine the numbers may vary. For example if you are using an 8K (K means thousand) machine, the Bytes Free number will be 7167. The flashing white square is called the Cursor and indicates your place on the screen should you begin to type.

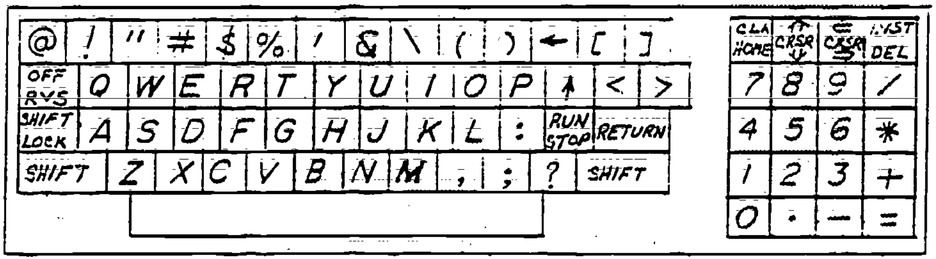

@ ! " # $ % ' & \ ( ) ← [ ]
OFF RVS Q W E R T Y U I O P ↑ < >
SHIFT LOCK A S D F G H J K L : RUN STOP RETURN
SHIFT Z X C V B N M , ; ? SHIFT
CLR HOME
↑ CRSR ↓
⇐ CRSR ⇒
INST DEL
7 8 9 /
4 5 6 *
1 2 3 +
0 . – =

Diagram of the Keyboard

* A series of operations stored in the computer's memory for the machine to perform is called a program.

B. The Keyboard

The keyboard is very similar to that of a typewriter, with several of the keys performing the same functions such as the space bar, shift and return keys. One main difference is that the PET prints in capitals. Holding down the shift key will print the symbol on the top half of the key. The following is a list of the special keys and their functions. Take time to experiment with them.

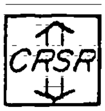
Shift: This key, when pushed with another key will print the top character on that key.

Shift Lock: Holds the shift key down.

Note: When the name of a key is followed by an asterisk "*" it indicates that the key is pressed with the shift key to obtain that operation.

Clear*: Will clear the screen (but not erase the memory).

Home: Will return the cursor to its home position--the upper left hand corner.



Cursor *: Will move the cursor up one line at a time--it must be used with the shift key.

Cursor : Will move the cursor down one line at a time

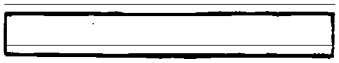
Cursor *: Will move the cursor one space to the left.

Cursor : Will move the cursor one space to the right.

Insert*: Will insert a space.

Delete: Erases one character at a time. The movement is from right to left.

Return: Will place the cursor at the beginning of the next line.



Space bar: Will create a space within a line. It will also erase characters from left to right.

RUN STOP

Run*: When pressed with the shift key, it is the command for loading a program.

Stop: Will stop a program.

OFF RVS

Reverse: Allows the reversal of the black and white portions of the characte To activate this mode, press the key unshifted. To return it to the off position, press the key along with the shift key. It can also be depressed to slow down the scrolling when a program is listed.

Before attempting the exercises which follow this section, do the examples on your machine. Press the keys as shown.

Press SHIFT + CLR HOME (this clears the screen)

Type: h o u s e

Press SHIFT + ⇐ CRSR ⇒ until the cursor is over the U

Type: R

This completes the correction of house to horse.

Type: U N T E D

Press SHIFT + ⇐ CRSR ⇒ until the cursor is over the T

Press SHIFT + INST DEL (this inserts a space)

Type: I

Press ⇐ CRSR ⇒ 3 times (this brings the cursor to the end of the word)

Type: S T A T L E S

Press SHIFT + ⇐ CRSR ⇒ until the cursor is over the E

Press INST DEL (this removes the L)

Press ⇑ CRSR ⇓ 7 times (this moves the cursor down 7 lines)

Press ⇐ CRSR ⇒ 4 times (this moves the cursor right 4 spaces)

Type: A M I C A

Press SHIFT + CRSR until the cursor is over the I

Press SHIFT + INST DEL 2 times (this creates 2 spaces)

Type: E R

Press SHIFT + ⇑ CRSR ⇓ 3 times (this moves the cursor up 3 lines)

Press SHIFT + ⇐ CRSR ⇒ 9 times (this moves the cursor left 9 spaces)

Type: O F

Your Screen should look like this:

```
UNITED

      STATES

            OF

              AMERICA
```

EXERCISE A

1. Type several lines of text, then clear the screen.

2. Type 'microcmputer'. Move the cursor, insert a space and type in the missing o.

3. Type Commoddore. Move the cursor to erase the extra d.

(Answer key is provided)

C. Using The PET As A Calculator

The PET can be used to solve mathematical problems. The symbols used for mathematic operations with the PET sometimes differ from those conventionally used. PET mathematic symbols are as follows:

/: The division symbol

*: The multiplication symbol

+: The addition symbol

-: The subtraction symbol

<: The less than symbol

>: The greater than symbol

<>: The not equal to symbol (use the two keys listed above)

↑ : The exponential, for example 5^2 would be typed on the PET as 5↑2

Since you will ask the PET to PRINT an answer, you must either type the word "PRINT" or use the symbol for print, which is "?". Thus to find the answer to 5+3 you must type: ? 5+3, then press the return key (or, PRINT 5+3). The screen will show:

```
? 5+3
 8
READY.
□
```

To divide 8 by 2:

```
? 8/2
 4
READY.
□
```

Ready, followed by the flashing cursor signifies that the machine is ready for another problem or command. (It is not a question.)

When solving arithmetical problems the PET follows the standard rules:

1. inside parenthesis are read first
2. exponential powers are calculated second
3. division/multiplication problems from left to right are solved next
4. addition/subtraction problems from left to right are solved last

The following problem would be read in this manner:

100-(2x(5+4)-1)+5↑2	5+4 is computed first (9)	100-(2x(9)-1+5↑2
	2 times 9 (18) -1 = (17)	100-(18-1)+5↑2
	5↑2=25	100-(17)+5↑2
	100-17 (83)	100-(17)+25
	83+25=108	83+25
		108

EXERCISE B

1. Solve these problems using the PET:

a) $\begin{array}{r}479\\ -362\\ \hline\end{array}$ b) $\begin{array}{r}367\\ \times\ 58\\ \hline\end{array}$ c) $493\overline{)79681}$ d) $\begin{array}{r}390012\\ -\ 64809\\ \hline\end{array}$

e) 5^4 f) 6x3+4÷2-1

2. How would the PET solve these problems?

a) 40-10/2

b) 3+15/3-1+5*2

(Answer key provided)

D. Running Programs

The major uses of computers in the classroom involves loading, running, and saving programs. The following section will explain the loading and saving operations first using a cassette tape recorder, then using a disk drive unit. Lastly, the running operation will be explained as it is the same for both the cassette and disk drive.

Loading A Program (Cassette)

There are different ways to load a program depending on the type of PET and how many programs are on the cassette.

METHOD A (does not apply to the PET 4000 series)

1. Place the cassette in the recorder and rewind it.
2. While holding down the shift key, press the run/stop key.

   The screen will print:

   ```
   PRESS PLAY ON TAPE 1
   ```

3. After you press the play button on the recorder the screen will print:

   ```
   OK
   SEARCHING
   ```

   Once a program has been located on the tape the screen will print:

   ```
   FOUND "NAME OF PROGRAM"
   LOADING
   ```

   When this method is used the first program found on the tape will be loaded. The program is ready to run when the screen prints:

   ```
   READY.
   □
   ```

   Some programs are written to begin automatically. For those that do not, your screen will look like this:

```
OK
SEARCHING
FOUND "NAME OF PROGRAM"
LOADING
READY.
□
```

METHOD B

1. Place the cassette in the recorder and rewind
2. Type LOAD then press the return key
3. The screen will print as in method A

This method ~~also loads the~~ first program found on the tape.

METHOD C

1. Place the cassette in the recorder and rewind
2. Type: LOAD "name of program" then press the return key

Note: The quotation marks must be typed and the name of the program must be typed exactly as it is titled on the tape.

3. The screen will print:

```
OK
SEARCHING FOR "NAME OF PROGRAM"
FOUND "NAME OF PROGRAM"
LOADING
READY.
□
```

This method is mainly used with cassettes containing more than one program. If a program is known to be near the end of a tape the fast forward button on the recorder can be used to approximate the location and save time.

EXERCISE C

1. Using various tapes try loading programs via each method.

Saving Programs (cassette)

To make additional copies of a program or to save a program you have written, the information must be stored on another cassette. To save a program which is loaded in the machine, follow these steps:

1. Place a blank cassette in the recorder and rewind
2. Type SAVE "name of program" and then press the return key

   (Note: Remember to include the quotation marks.)
3. The screen will print:

   ```
   PRESS PLAY AND RECORD ON TAPE 1
   ```
4. Press both buttons simultaneously
5. The screen will print:

   ```
   OK
   WRITING "name of program"
   ```

Depending on the length of the program, the writing process may take from 30 seconds to 5 minutes. Upon completion the screen will print:

```
READY.
□
```

To ensure that the program was copied correctly, it is wise to verify the accuracy. To verify a program follow these steps:

1. Rewind the tape.
2. Type VERIFY then press the return key
3. The screen will print:

   ```
   PRESS PLAY ON TAPE 1
   ```
4. After pressing the play button the screen will print:

   ```
   OK
   VERIFYING
   READY.
   □
   ```

(Note: It will take the same amount of time to verify as it did to write the program. The word ready will appear upon completion of the process.)

If the screen prints: VERIFYING ERROR? repeat the saving process from Step 1 through the verification process. If the problem continues try cleaning the cassette recorder (this process is discussed on page 18). If that fails, try using a new blank cassette. Do not worry about losing your program. It will remain in the machine until the power is turned off or you type 'NEW' followed by the return key.

EXERCISE D

1. Practice making copies and verifying tapes. (Use only public domain tapes.)

Loading a Disk: 3.0 Basic

At first glance the loading of a disk looks complicated. However, after doing it a few times you will find that it is quite simple. Before actually loading a disk there are a few things you should know first.

a) Before turning on the equipment make sure there are no disks in the unit.

b) Always turn on the PET first, then the drive unit.

c) Always turn off the drive unit first, then the PET.

d) Always hold the disk by the label--never touch exposed portions of the disk.

e) The disk is always entered with the label facing up and the square notch on the left.

Now you are ready to load a disk.

1. Turn on the PET, then the disk drive.
2. The red indicator lights will come on briefly, then turn off.
3. Choose a disk and gently slide it into one of the slots and close the door.
4. Type: OPEN 1,8,15 (this must be typed every time a disk is loaded).
5. Type: PRINT#1,"I drive no." Your drive no. is either 1 or 0. Check which slot your disk is in.
6. Type: LOAD "$ drive no.",8 This loads the directory so you can see what programs are available.
7. Type LIST to view the directory.
8. To load one of the programs type:

```
LOAD"drive no.:program name",8
```

The program name must be typed in just as it appeared in the directory.

EXAMPLE: A game disk is put into the drive 0 slot. To load a program here is the sequence of steps:

```
OPEN 1,8,15
PRINT#1,"IO"
LOAD"$O"
LIST (to see directory)
LOAD"O:Aliens",8
```

Saving Programs (Disk): 3.0 Basic

Saving programs is done very quickly with the disk drive. Before this is done a diskette must be prepared or formatted. The steps for preparing a diskette are as follows: (drive 0 is used here as an example)

1. Type: OPEN 1,8,15
2. Type: PRINT#1,"NEWO:disk name,23" (any 2 digit number can be used)
3. To check the formatting load the directory

Now the diskette is ready for copying. For this example drive 0 contains the blank diskette and drive 1 contains the diskette with programs.

1. Load a program to be copied from drive 1
2. Type: SAVE"0:name of program",8
3. To verify type: VERIFY"0:name of program",8

To duplicate an entire disk follow these steps:

1. Prepare a diskette as described above.
2. Type: PRINT#1,"DUPLICATE 0=1"

In this example the blank diskette is in drive 0 and the diskette containing the programs is in drive 1. The drive containing the blank diskette always comes first. DO NOT CONFUSE the drive numbers or you may lose all of your programs. Verify as described above.

Erasing Programs (Disk): 3.0 Basic

You can erase individual programs from a disk or serveral at once. To erase 1 program type:

```
PRINT #1,"S drive no:name of program "
```

To erase several programs, where the names are not similar, type:

PRINT#1,"S drive no: name of program, drive no:name of program..."

Example: PRINT#1,"SO:Aliens,O:Hangman"

To check if your programs have been erased simply list the directory. Those programs should no longer appear.

Using a disk: 4.0 Basic

Using the 4040 dual disk drive is even easier then using the 2040 disk drive. There is no need to open or initialize during its operation. Follow these steps: (drive 0 is used here). Insert the diskette into drive 0.

To load the directory:

1. Type: LOAD"$0",8

Loading Programs:

1. Type. LOAD"0:name of program",8

To prepare a new diskette before copying:

1. Type: HEADER"disk name",D0,I99 (any 2 digit number can be used in place of 99.)

After this has been typed the computer will ask "Are you sure?" Be sure you have indicated the correct drive number that contains the blank diskette.

Saving Programs:

1. Type: SAVE"0:name of program",8

To verify a program:

1. Type: VERIFY"0:name of program",8

To copy an entire disk: (This command prepares the diskette.)

1. Type: BACKUP D0 to D1 Where drive 0 contains the diskette of programs and drive 1 contains the blank diskette.

Erasing Programs:

To erase an individual program from a disk.

Type: SCRATCH D0,"name of program"

Running a Program

Running a program means just what the name implies--putting a program into operation. In other words having the PET "run" the program.

It is important to remember to answer specifically the questions the PET asks and to watch the screen to determine if the return key is necessary.

To begin a program that does not do so automatically after loading, type: RUB. Proceed to answer the questions printed on the screen. Most programs accept Y or N for yes/no answers.

Some programs require that you press the return key after each response to a question. Watch the screen. If nothing happens after you type in your response, press the return key. If you press it unnecessarily you will break out of the program, that is, the program will stop and the screen will print:

```
READY
```

To intentionally terminate the program before completion press the run/stop key. This key is inactivated, however, if the PET is waiting for your response. In that case, type any response then immediately press the run/stop key. If for some reason that fails, press the shift and run/stop key (as if to load). Just remember that no matter what combination of keys you press, you will not harm the machine. When typing the run/stop key the screen will print:

```
Break in 'A Number'
Ready
```

If you accidentally break out of a program and wish to continue with the program you must type CONT then hit return. The PET will then print a question mark and wait for you to answer the last question asked. (Note: You must not type any other keys before typing CONT or the PET will not allow you to return. Also some programs, especially those using graphics will not function correctly when typing CONT.)

If you break out of a program and wish to start over again type run then return.

If after answering a question the PET prints:

```
? Redo from Start
?
```

The PET probably asked a question with a numeric answer to which you responded with a letter, word, or symbol. Reread the question then type the appropriate response to continue.

E. Cleaning and Caring for Your PET

The PET should be cleaned approximately once per month. If a great deal of copying is to be done, it should be cleaned prior to and during the process to give better results (clean every 10 tapes).

The cleaning process is very simple and takes less than 5 minutes. Rubbing alcohol and Q-tips are needed.

Follow these steps:

1. dip the Q-tip in the alcohol
2. press the reject button on the tape recorder to open the case
3. press the play button
4. hold the Q-tip lightly against the rotating drive wheel
5. change Q-tips until no dirt appears on the cotton
6. wipe the metal recording head
7. wipe the inside to rid the cassette of all dust

(Note: Check the head, drive wheel to be sure no cotton from the Q-tip remains within the machine.)

EXERCISE F

1. Clean your PET cassette
2. Load and run a program

(If a load error appears, check the tape recorder for cotton fibers and attempt to reload.)

F. Editing

In order to alter existing tapes to meet individual needs it is helpful to understand a little about programs and a few BASIC commands. Type the following program into your PET. Be sure to type NEW (followed by the return key) to clear the memory. After typing each line be sure to press the return key.

```
10 Print "Hi there"
20 Print "I'm your PET"
30 End
```

This is a short program which will print two simple statements on the screen. Notice, each command is preceeded by a line number. The line numbers tell the PET in which order to perform the operations. Now clear the screen (press shift and clr/home) and run the program. Your screen should look like this:

```
HI THERE
I'M YOUR PET
READY.
```

Clear the memory again and type in the following program:

```
10 Read A$, B$, C$
20 Print A$, B$, C$
25 Go to 10
30 DATA cat, dog, house
40 DATA run, see, it
50 End
```

Now run the program. Your screen will look like this:

```
CAT   DOG   HOUSE
RUN   SEE   IT
OUT OF DATA ERROR
```

Line 10 instructs the computer to read the information stored in the Data statements. Line 20 instructs the computer to print the information on the screen. Lines 30 and 40 store the information that is to be printed. Line 50 ends the program. (do not concern yourself with line 25)

Notice that in line 10, three variables are used and in lines 30 and 40 (called the DATA statements) three words are used. Thus whenever you want to change information in a program the variables determine how many pieces of information can be stored in the DATA statements.

The $ after the letters A, B, and C tell the computer that words (or letters) will be found in the DATA statements. If you want to store numbers, simply omit the $.

The statement 'Out of Data Error' simply means the PET ran out of stored words to print after line 40. In commercial programs there is a command to eliminate that statement, but one that you need not know to use the program.

When altering programs the changes are made in the DATA statements. To edit a program follow these steps:

1. Load a program (such as tachistoscope)
2. Run the program so you are familiar with its objective
3. Break out of the program and type LIST
4. Remember--to slow down the scrolling, press the off/rev key
5. When the list of DATA statements appears, release the off/rev key and press the run/stop key
6. Type your words or phrases over the existing text in the DATA statements. Be sure to seperate each word with a comma. Erase all extra characters (use the space bar).
7. When the line is completed, press the return key to register the change.
8. Continue through the list making sure that the line lengths do not exceed the limit defined in the program, or the number of words is the same as the original program.
9. Run the new program to check for errors.
10. Save the new program.

EXERCISE G

Choose two of the following programs to create your own personalized tapes. Be sure to follow all of the above steps. Flash, U-do-it-spell, Hangman, Vocab-u-pet.

G. Helpful Hints

1. If the room has static do encourage students to touch another object before touching the PET to avoid freezing the machine.

2. Do keep the PET as free of dust as possible.

3. Do unplug the PET overnight, vacations and weekends.

4. Do wait at least 1 minute after turning off the power, before turning it on again.

5. Do not allow food or drink near the machine.

6. Do not place near a heat source or in front of drafty windows.

7. Do not leave the play button depressed.

8. Do not use cassette-type tape head cleaners.

TROUBLE SHOOTING

Correcting Error Messages

Occasionally error messages will appear on your screen, or a program may stop in the middle. Most often they are the fault of the user. Here are some common errors and the correction procedures:

Problem	Cause	Correction
1. The PET prints: SYNTAX ERROR	misspelled command word command not in BASIC. extra symbol included	correctly type the word use a legal command delete the extra symbol (usually a comma or period)
2. program breaks out by itself	user did calculations and omited the ? while a program was in the memory	delete the problems that were printed by listing the program
3. program doesn't respond to commands	shift lock is engaged poor wiring froze program static electricity	release the shift lock shut off machine, reload program
5. The PET prints: LOAD ERROR	faulty tape dirty machine	rewind tape, reload clean tape heads
6. incorrect graphics displayed	program is in wrong case mode (either upper or lower)	to change upper to lower case type: POKE 59468,14 to change lower to upper type: POKE 59468,12

There are other errors and causes which can occur during the operation of the PET. If you are not sure of how to correct the problem, try returning the program. If the problem persists, seek the assistance of someone more familiar with the operation of the PET. Most corrections are relatively simple and will become routine with experience on the machine.

III. SOFTWARE APPLICATIONS

There are five basic types of computer programs each serving a specific purpose. The majority of instructional programs are drill and practice or simulation. When selecting programs for a student, keep the objective for its use in mind.

1. Drill and Practice: As th. name implies, this program gives the student repetitious activities to reinforce a particular skill. Basic Facts (public domain), You-do-it-spell (public domain) and Metric Est (Cursor #2) are but three examples.
2. Tutorial: This program actually provides so.e instruction to a concept or skill. Examples are given and usually followed by a practice set of activities. Uses of homonyms and grammar rules would fall into this category.
3. Simulation: This type of program is frequently perceived as a game by students. It presents a particular situation or event and allows the student to make decisions which affect the outcome. Ruling a country Kingdom (Personal Software) and running a business--Lemonade Stand (public domain) are two examples.
4. Logic: This type of program requires the student to use inductive and deductive reasoning skills. In effect these programs teach students how to problem solve by presenting a puzzle which allows many avenues to be taken to arrive at a solution. Hanoi (Cursor #5) is such a program.
5. Games: These programs were designed to be purely for entertainment purposes, however many also require certain skills to be successful. For example, Aliens (Cursor #16) requires astute eye-hand coordination and Battleship (Pet cassette exchange) requires the student to plot a point on a grid.

IV. CLASSROOM USE AND MANAGEMENT

To enable the effective use of micro-computer systems within the classroom, the user should have knowledge of:

A) Component system hookup
B) Physical orientation of the computer
C) Classroom Management
  1. scheduling
  2. contract and reward systems

A. Component System Hookup

Although the PET can function as an isolated unit, when connected with peripheral devices the versatility of the PET is greatly enhanced, providing for a wide range of classroom applications.

Common peripheral devices available for operation with the PET are: cassette tape recorders, dual disk drives, and printers.

The cassette recorder has its own cable and connector attached. There is only one slot which it will fit into, and it fits only one way. The slot is located in the left rear corner (facing rear) of the PET.

If you are using one peripheral--either the disk drive or the printer--you will need a PET to IEEE cable. The large flat end connects to the PET in the large right hand slot at the rear of the machine (facing rear). The other end slips into the only slot at the rear of the printer or disk. Be sure to tighten the screws.

If you are using both a printer and a disk and IEEE to IEEE cable is needed. First connect one of the units to the PET as described above. Then connect one end of the IEEE to IEEE cable to the other unit and piggyback (by screwing on the connector) the last end to the connection on the first unit. Both the disk drive and printer must be plugged into an outlet.

IMPORTANT: 1. NEVER connect or disconnect any peripheral while any of the units (including the PET itself) is turned on.

2. Be sure to instruct your students (if more than one PET is hooked up to the peripherals) not to request the services of a peripheral at the same time. (Example: Do not have two students requesting the printer to print an exercise at the same time.)

3. DO NOT use a three prong adapter. Three hole outlets MUST be used.

B. Physical Layout of Hardware

When deciding how to arrange your equipment within the classroom, several important items must be considered:

1. Attempt to locate all equipment in the same general area of the room.

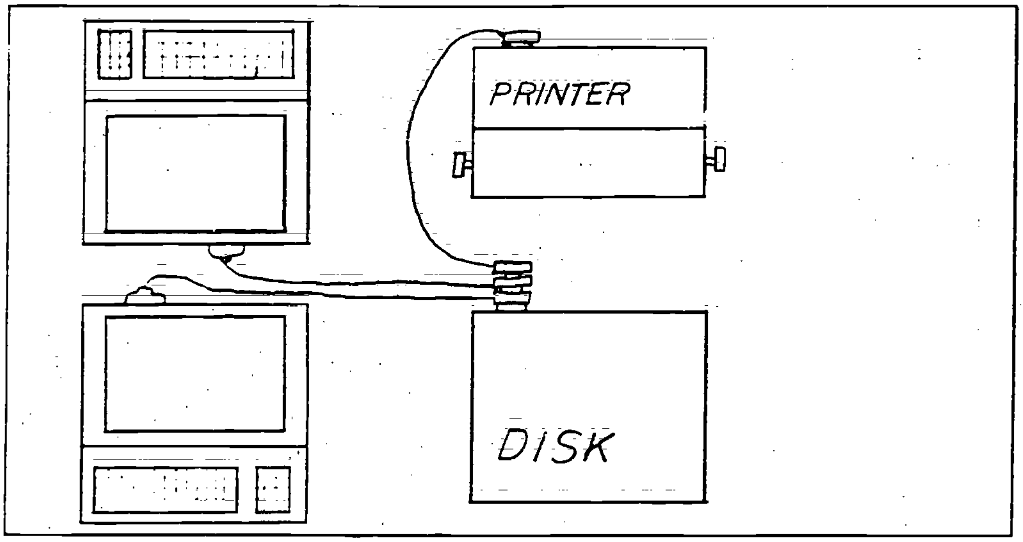
2. Do not locate the equipment in drafty or dusty areas.

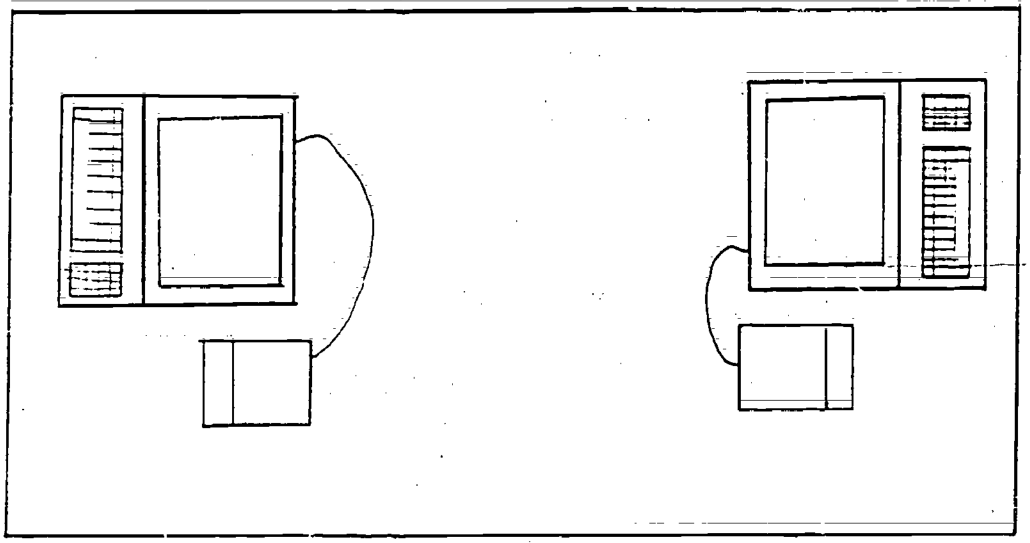
3. If possible isolate the equipment from the rest of the room. (The machines are highly motivating and can be distracting to other students.) If isolation is not possible attempt to face the screens away from the general instrucional areas.

4. Each computer and peripheral device (except cassette recorders) will require an electrical outlet.

5. If multiple computers and peripherals are to be used together, they can be placed no further apart than their cables will permit.

On the following page you will find suggested layouts for the equipment.


PRINTER
DISK



C. Classroom Management

To effectively use computers in the classroom it is helpful to construct a usage chart and post it in the room. This helps both the students and the teacher see, at a quick glance, whose turn it is to use the PET. The chart should specify the program to be used and the amount of time to be spent on the program. A suggested time is a minimum of 20 minutes if a cassette is used. If a contract or reward system is to be used, set aside a block of time each week for this purpose.

It is most important to instruct students in the use of the computer equipment, before allowing them to use it. The student lesson book provided can be used individually or with a group. Be sure to cover classroom rules concerning the equipment and stick by them.

The two charts which are found on the following pages are suggestions as to how computer time may be allocated. They are based on a 15 student caseload. The first chart allocates time using the students as a base, the second chart uses subject matter. Additionally, examples are given of contracts used as a reward system.

WEEKLY COMPUTER SCHEDULE

Week of ______________

HOUR	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
1	Math Simulations				
2	Reading Logic				
3	Math Science				
4	Reading Social Studies				
5	Special Purposes				
	Teacher Preparation Time				

WEEKLY COMPUTER SCHEDULE

Week of ____________

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
ob A. 8:00-8:20	Mult.	basic Facts	Mult.	MAKE
im C. 8:20-8:40	Div.	Darts	Div.	UP
ohn L. 8:40-9:00	Add	Drag	Add	TIME,
Sue B. 9:00-9:20				
Jim M. 9:20-9:40				SPECIAL
Pat R. 9:40-10:00				
Tom O. 10:00-10:20				USES
Ed P. 10:20-10:40				
Gil R. 10:40-11:00				and
Pam S. 11:00-11:20				FREE
Bob F. 11:20-11:40				
Bill T. 11:40-12:00				
Tom N. 1:00-1:20				TIME
Fred B. 1:20-1:40				
Eric L. 1:40-203				
Teacher Conference Period				

CONTRACT EXAMPLE #1

(student name) will complete (assignment) by (date). with (accuracy). In return he/she will earn (minutes) of free time on the computer.

John Doe will complete 10 pages in Math by September 15 with 80% accuracy. In return he will earn 20 minutes of free time on the computer.

CONTRACT EXAMPLE #2

(student name)____ will ______(minutes)______ of free time on the computer when he achieves ___(% mastery)______ on ______(assignment)______.

John Doe will earn 30 minutes of free time on the computer when he achieves 90% on his spelling list.

CONTRACT EXAMPLE #3

This system can also be used with EI students where specific behaviors are substituted for assignments and used to reward students.

____(student name)__________ ill earn minutes of free time on the computer when he/she _____(behavior)____________ _______(frequency)____________.

John Doe will earn 45 minutes of free time the computer when he completes all assigned work 4 days in a row.

V. EVALUATING SOFTWARE

When using computer programs with students for remedial purposes there are several aspects to consider. Many programs are appealing on the surface, but a closer look can reveal characteristics which make them inapporopriate for remedial use. The following is a checklist of items to consider when selecting instructional programs:

SOFTWARE IN GENERAL

- Does the material make realistic assumptions about the user?
- Are instructions available, and if so, are they clear?
- Is support material necessary, and if so, it it available?
- Has the material been field tested and/or otherwise evaluated?
- Is there documentation for the instructor to use in modifying the material?
- Is the text on the screen readable (presented in a concise uncluttered fashion)?
- Is the readability level of the text appropriate for the target user?
- Are response modes designed for the target user?
- Does the use of sound, graphics, color, etc. distract from the lesson rather than enhance?
- Are capabilities of the microcomputer fully and imaginatively used in the lesson? (color - graphics - sound)
- Is there evidence of fun while learning? (motivation)
- Does the program use putdowns?

DRILL AND PRACTICE/TUTORIAL

- Is the material contained in the lesson consistent with current practice?
- Does the lesson exist in isolation or is it part of a collection of modules and support material?
- Can the student make an entry into the lesson other than the beginning?
- Is there evidence of internal recordkeeping - pacing the user through the lesson?
- Does the material provide the instructor with a record of student responses?

SIMULATION

- Does the simulation clearly state its objective?
- Is the simulation representative of the event it purports to image?
- Can the simulation be re-entered at a point of exit?

EXERCISE H

1. Evaluate these programs: You-do-it-spell, Crossbow, Guess-my-sentence using the above criteria.

SOFTWARE GUIDE

The following pages are an excerpt from the software guide used within the Linden, Fenton, Lake Fenton Compu r Consortium. The software guide should include a listing of available software within a district as well as a brief discription of each program to assist interested teachers in the selection of appropriate software.

PET Language Arts Programs

Alpha Man--An exercise which teaches the alphabet. The user must supply missi letters of the alphabet.

Alphabet Search--A tracking exercise using letters. The time required is reco

Alphabetizing--The user alphabetizes two words at a time. A correct response produces moveable graphics.

Dolch 1-11--The flash program using Dolch words.

Flash--Can be used to teach new words or spelling. Twenty words flash one at time at various places on the screen. The user is to type them. The per cent correct is given at the end of the program. The word list can be changed.

Guess My Sentence—Contains two tapes, elementary reading exercise. A repetiti drill and practice, using scrampled sentences (basically three letter word ten sentences per series.

Hangman--Computerized version of this old favorite. Super graphics. Word lis can be changed.

Jotto--A guessing game involving 5 letter words. The PET will tell you what letters you have guessed correctly.

Kucera 1-4--The flash program using the Kucera Francis words.

Madlib--A computerized version of this oldie but goodie. The user is asked to type in verbs, adverbs, etc. A paragraph then appears using the supplied words.

Match--The game of concentration using symbols.

Phrase--A game for two players. It is like hangman but uses phrases or senten It can be used for sentence structure or fill in the blank practice as in test review.

Q's and Z's--Clues are given where the correct answer begins with a Q or Z.

Scrambler--An alphabetizing exercise using letters. The rating at the end of sequence is based on speed. (not as easy as it sounds)

Search-- program which creates word search puzzles. A printer is needed.

Spell--This program allows you to create word lists of your own. (five lists which can each contain 26 words). The user listens to taped words and spells them on the PET. It then lists words correctly spelled and those needing practice.

Spelling Drill--A spelling exercise in which the user types in words to practice. Words appear on the screen with their letters scrambled. The number of words practiced, number spelled correctly, help needed, and answers given are displayed.

Tachistoscope--An exercise to increase sight vocabulary. Words or phrases are briefly flashed on the screen for the student to read and retype.

Vocab-U-Pet--A spelling program. To put in your own list of five words at a time, type 'teacher' when the PET asks you your name. Directions will be given. The PET then will flash the words one at a time and the user must type them.

Word Machine--Elementary reading exercise. The user looks for a word printed on the screen among foils. Three letter words are already in the program or the user can type in own three letter words. Score is computed upon completion.

Write to Read--An elementary reading exercise. The user reads an types missing words in senetences (three letter words). Gives time and number correct and incorrect. Four tapes.

You-Do-It-Spell--A program which allows the user to create 5 word lists containing 25 words each for spelling practice. A standard tape recorder is needed.

PET Math Programs

Arrows--A simulation of shooting arrows at a target. Angle and velocity are to be calculated.

Basic Facts--Flash drill of basic math facts. Calculates number of correct responses under 4 seconds, number of correct responses over 4 seconds, and number of correct responses untimed.

BJack--The game of blackjack. (Good for mental computation.)

Box--A game in which the player finds atomic deflection patterns on a grid. Uses coordinates and some elementary angle geometry. (Must type run 1 to run the program.)

Chemist--A game in which the player attempts to dilute solutions without blowing up. Provides practice with ratios and proportions.

Crossbow--A game in which the user tries to determine the fractional representation of a dot on a scale. Excellent practice determining the relationships between fractions. Three skill levels are provided.

Darts--Math game to increase mental computation skills in addition, subtraction, multiplication, and division. Can be used with one or two players. Three skill levels are provided which correspond to the number of digits in a problem.

Drag--A graphic drag race for 1 or 2 players. Simple addition and subtraction problems are given. Correct answers move the cars.

Edutilities--Calculates the greatest common divisor, surface area of a cylinder, triangle information, base number conversion, temperature conversion, prime number generator, Roman numerality, and metric conversion.

Equiv Fractions--A tutorial program teaching the concept of equivalent fractions.

Financial Math--Computes calculations in business and money, trigonometry, probability and statistics, and a conversion table. Each category has several subcategories to chose from.

Grid--A short tutorial and drill on plotting coordinate points on a plane.

Kids Algebra--Teaches the basic algebraic principles using three variables and addition and subtraction. The program begins with simple one digit problems. Success on a ten problem series determines the next level of difficulty. (Moves toward multi-digit problems)

Large/small frac--A drill and practice whereby the user selects the larger of two fractions presented. Options include proper fractions only or proper and improper fractions.

Lemondate Stand--A simulation of operating a lemonade stand as a business for profit.

Long Division--Teaches the skills of estimating in long division. Nine grade levels are provided. When incorrect the correct solution is shown with enough time for the student to find his error. Calculates the number correct upon completion of the program.

Martian Math--An addition drill in the form of: 6 tens plus 15 ones. Graphics display with a score of 100%.

Math--The game of concentraion--using numbers. The catch is to match numbers adding up to a specified sum.

Math Whiz--A timed drill for 2 players. A problem with answer is presented. The players must decide if the answer is correct. A running score is kept. Addition, subtraction, multiplication and division are offered as options.

Maxit--A game for one or two players. A grid with positive and negative integers is shown. Players take turns trying to capture numbers. Each player can move in only one direction. The score is computed after each turn. Teaches the addition of positive and negative numbers.

Metric Conv--Helps students (through the use of a chart) how to convert one metric measure to another.

Metric Est.--An exercise whereby the user is presented with graph[illegible] or areas and is asked to estimate their size using centimeters. [illegible] correct is computed upon completion of the program.

Mix/Improp Fractions--A drill to change mixed numbers to improper fractions and vice versa.

Plot--A program graphing functions; sin, tan, and the user can put in his own function.

Snoopy--A program teachi[illegible] addition of positive and negative numbers. A number line presents problems. Correct answers count as shots at the Red Baron (and a chance to see him parachute out of the sky). Wrong answers are shown as bullet holes in Snoopy's doghouse. Five skill levels correspond to the time provided for a response.

Red/Equiv Frac--Drills students on either reducing fractions to lowest terms or completing equal fractions.

Ruler--Graphically aids students in reading ruler. Corrections to incorrect responses are shown graphically.

Tictacarith--Tic Tac Toe using math problems.

Tic Tac Pet--Tic Tac Toe using equations. Ten skill levels are provided involving negative and positive intergers, decimals etc.

Train--The program presents story problems involving time, speed, and distance.

PET GAMES

Aliens--Shoot the aliens before they get you.

Ambush--Isolate your enemy in a forest.

Bagels--A guessing game. Try to guess a four digit number. Exact means right number in right sequence. Matches means right number, wrong place.

Bat--Keep your bat alive by feeding him.

Battleship--Computerized version of the find the ship on the grid game.

Bjack--The game of Blackjack.

Boswain--Which hand has the gems? Don't lose your fingers!

Brick--A game in which the user attempts to estimate the speed of a moving (which disappers) and stop it as close to a window as possible without going through.

Capture!--Trap the beasts before they get you.

Catch--Play catch with the PET.

Chase--A game for one or two players requiring quick reflexes and coordination. One player tries to catch another or the PET by moving symbols on a grid. The time elapsed is displayed upon capture. Watch out for the trap doors. They put you anywhere on the screen.

Checkers--An old favorite computerized.

Cops--A two car chase with cops and robbers.

Defend--Defend your space station from attackers.

Dots--The old favorite computerized.

Draw Poker--Computerized version of the same game.

Dungeon--The game of dungeons and dragons.

Escape--A game played against the computer that calls for quick thinking. You try to escape from a prison patrolled by robots. Moves are counted upon completion of the program.

F Ball--A game of football played against the PET or an opponent. It may be helpful to write down the plays and corresponding numbers for reference during the game.

Fifteen--Move the numbered squares around to get them in order.

Fire--Put out the fire before you run out of time.

Frog--Keep your frog alive by catching flies.

PET GAMES

Gammon--The game of Backgammon played against the computer. Before loading this program type "NEW" to clear the memory. (This program requires the entire capacity of the PET).

Godzilla--Save the town from the dreaded monster.

Gomoku--Similar to Othello. 5 levels of play.

Joust--A simulation of the medieval sport of jousting.

Kalah--An old game using rocks in a pit.

Miner--Dig for gold.

Nab--Similar to a pin ball game. Get as many points as you can before you crash.

Ouranos--Destroy your enemy's house using nature.

Poker--A game of poker against the machine. The PET calculates who owes whom and how much.

Police--Try to catch a criminal using a map and clues.

Race--A two player game. User picks the speed and the number of pylons then races opponent to the finish line.

Rail--Run a rail station without crashing your trains.

Rat Run--Find your way out of a maze.

Reverse--Similar to the game of Othello.

Road Race--A maze game. Try to maneuver a dot through a maze in a race against time. Skill levels of one to nine correspond to the speed of the moving dot. Sound easy? It's not. A real test of reflexes.

Shark--A game in which you are the shark and attempt to get as many swimmers as possible and avoid the divers before the people can swim to shore.

Sheep--Get your sheep in the barn before they eat all the corn.

Shoot--A game for quick reflexes. Try to shoot a moving target. Five skill levels correspond to the speed of the target.

Snake--A game to play against the PET or a friend. Tests quick reflexes and coordination. The object is to wind your snake around without running into a wall or yourself.

Spot--Tic Tac Toe but you must get 4 in a row.

Thunt!--Find the treasure before the robots get you.

Yahtzee--Computerized version of the dice number game. Up to four can play.

PET Social Studies Programs

Capitals--A drill which teaches the capitals of states and countries. The user may choose between multiple choice or fill in the blank for response.

Depth Charge--A game which teaches directionality (North, South, East, and West). The player must locate a mine on a three dimensional grid before it blows up.

Hurkle--A game which teaches directionality using two dimensions.

Kingdom--A simulation game where the user rules the land of Sumeria. Land is sold, crops planted, and people are fed through a series of decisions. The effectiveness of ruling the country is calculated upon termination of reign.

Westward Ho--A simulated trek across the country where many decisions must be made along the way. Very difficult to succeed from the start. (If you answer "yes" to the question "Do you want to play again?" just after you have died on the trail it will begin the next game on the spot where you died.)

PET Science Programs

Box--See Math

Chem [illegible]--See Math

Edutilities--See Math

Metric Est.--See Math

Space Facts--A program which allows you to travel to distant planets to find out how much you weigh, how high you can jump and how far you can throw a ball on those planets.

PET Logic Programs

Dots--A computerized version of the paper and pencil game. The user competes against the computer.

Hanoi--An excellent game of logic. The user attempts to move an entire stack of blocks from one pad to one of two others, never putting a larger block on top of a smaller one. This is to be done in the fewest possible moves. You may choose a pile from three to seven disks. Upon completion it will state the number of moves you took and the fewest moves possible.

One Queen--A game based on the move of the chess queen. Very difficult--I don't know if it's possible to beat the computer. Can you find a way?

Revers--The game of Othello. The object of the game is to get as many of your symbols on the board as possible. When you capture an opponent his symbols reverse to yours. To capture you must have a symbol on both sides of your opponent (They don't tell you that in the directions) You may compete against another person or the computer. It can be beaten.

VII. ANSWER KEY

Exercise I

1. press shift and clear/home key

2. move the cursor left until it is over the M, insert a space by pressing the shift and inst/del key, type in the letter O

3. move the cursor left until it is over either d, press the delete key to erase the extra d.

Exercise II

1. a) 841 b) 21286 c) 161.624746 d) 325203 e) 625
f) 19

2. a) 35 b) 17

VIII. GLOSSARY

1. ASCII - American Standard Code for Information Interchange. Assembly language.

2. BASIC - Beginners all-purpose Symbolic Instruction Code. The language you would use to program the PET.

3. Bit - A piece of information in base 2.

4. Byte - A standard unit of memory. There are 8 bits to a byte.

5. Chip - a microprocessor.

6. CPU - Central Processing Unit. The circuit board.

7. CRT - Cathode Ray Tube display, the video screen.

8. Cursor - The flashing white square denoting placement on the screen.

9. Data Statements - A list of constants to be read by the computer. The contents of these statements can be controlled by the user; example, spelling lists.

10. direct mode - The mode in operation when the PET is turned on. The PET can be used as a calculator or to print text in capital letters when in this mode.

11. Floppy disk - A flexible disk for storing data which has a high speed retrieval.

12. Graphic mode - The mode in operation when the shift key is depressed. To obtain lower case letters type POKE 59468,12.

13. List - The command which will produce the actual program in BASIC language. This command is used when a change in data statements is desired.

14. Memory - The storage capability of a computer. It is measured in K's where 1K-1024 bytes.

15. Motherboard - A circuit board containing empty slots which allows future expansion of memory or the addition of peripherals.

16. New - This is a command which clears the memory.

17. Peripherals - Adjuncts to the computer; i.e., printer, floppy disk, etc.

18. Poke - A statement to put the PET into graphic mode and back to direct mode.

19. RAM - Random access memory, the board containing data in binary code.

20. ROM - Read only memory - permanent storage of data that cannot be changed; i.e., the BASIC language.

IX. BIBLIOGRAPHY

The following is a list of helpful books and magazines for further information.

Books

1. Coa¬, James S., Basic Basic An Introduction to Computer Programming in Basic Language, Hayden Book Co., Inc., Rochelle Park, New Jersey, 1978.

2. Coan, James S., Advanced Basic Applications and Problems, Hayden Book Co., Inc., Rochelle Park, New Jersey, 1977.

3. Donahue, Carroll S. and Enger, Janice K., PET/CBM Personal Computer Guide, Osborne/McGraw-Hill, Berkeley, Ca., 1980.

4. Dwyer, Thomas and Critchfield, Margot, Basic and Personal Computer, Addison Wesley Pub. Co., Mass., 1978.

5. Osborne, Adam and Donahue, Carroll S., PET/CBM Personal Computer Guide Second Edition, Osborne/McGraw-Hill, Berkely, Ca., 1980.

6. Simon, Daved E., Basic From the Ground Up, Hayden Book Co., Inc., Rochelle Park, New Jersey, 1978.

7. Weizenbaum, Joseph, Computer Power and Human Reason, From Judgement to Calculation, W. H. Freeman and Co., San Francisco, 1976.

Magazines

1. Byte
70 Main Street
Peterborough, N.H. 03458
$15/year

2. Classroom Computer News
P.O. Box 266
Cambridge, Massachusetts 02138
$9/year

3. Compute! Magazine
Box 5406
Greensboro, North Carolina 27403
$20/year

4. Creative Computing
Box 789-M
Morristown, New Jersey 07960
$15/year

5. Kilobaud Microcomputing
80 Pine Street
Peterborough, New Hampshire 03458
$25year

6. MACUL Journal
Larry Smith WCISD
33500 Van Born Road
Wayne, Michigan 48184
$5/year

7. Micro
P.O. Box 6502
Chelmsford, Massachusetts 01824
$18/year

8. Personal Computing
1050 Commonwealth Avenue
Boston, Massachusetts 02215
$12/year

9. Recreational Computing
1263 El Camino Real
Box E
Menlo Park, California 94025
$10/year

10. The Journal
P.O. Box 992
Acton, Massachusetts 01720
free on limited basis or $15/year

APPENDIX A

USAGE LOG

Student ______________________ Date ___________ Classroom ___________

Date	Program	Type*	Total Time

* M-Math L-Language Arts G-Game

APPENDIX B

for use with 3.0 disk

1. Load the Directory

```
Open 1, 8, 15
Print #1, "IO"
Load "$0", 8
List
```

2. Load a Program

```
Open 1, 8, 15
Print #1, "IO"
Load "0: Hangman", 8
```

3. Prepare a New Diskette

```
Open 1, 8, 15
Print #1, "NEWO: Math Disk, 23"
```

4. Save and Verify a Program

```
Save "0: Hangman", 8
Verify "0: Hangman", 8
```

5. Copy an entire Disk

```
Print #1, "Duplicate 0 = 1"
   (0 is the blank diskette)
```

6. Erasing A Program

```
Print #1, "SO: Hangman"
```

APPENDIX C

for use with 4.0 disk

1. Load the Directory

```
Load "$0", 8
```

2. Load a Program

```
Load "0: Hangman", 8
```

3. Prepare a New Diskette

```
Header "Math Disk", DO, I22
```

4. Save & Verify a Program

```
Save "0: Hangman", 8
Verify "0: Hangman", 8
```

5. Copy an entire Disk

```
Back up DO to D1
  (D1 contains the Blank Diskette)
```

6. Erasing a Program

```
Scratch DO, "Hangman"
```

SPECIAL EDUCATION COMPUTER MANUAL

Student's Manual

developed under Federal Grant #G008100281

Model Program in Microcomputer Utilization with Handicapped Students

by Paulette M. Hummel, M.A.
Fenton Area Public Schools

and William R. Hummel, M.A.
Linden Community Schools

TABLE OF CONTENTS

INTRODUCTION

What follows is a manual to teach students how to use the PET computer. The first section is mandatory for all students. It is divided into six lessons:

Lesson I	Turning on the Machine
Lesson II	Using the PET as a Calculator
Lesson III	The Keyboard
Lesson IV	Special Keys: The Space Bar and Return Keys
Lesson V	Loading a Program
Lesson VI	Running a Program

Also included is a list of do's and don'ts.

Following each lesson is a quiz or exercise. Be sure your students complete them. After they have completed the first section and passed the review quiz, you should then instruct them in some simple correction procedures should a problem occur or if typing errors are made while using the PET. Topics to be covered are:

Error Messages

1. syntax error
2. break message
3. redo from start
4. load error

How to Erase Mistakes

1. the del key
2. the cursor left key
3. The space bar

After receiving this instruction, students who want to learn more about this machine can work through the optional second section covering the cursor control keys. It is designed for the students to complete independently.

LESSON I

How to Turn the PET On

Look at the back of the PET. In the lower right corner there is a switch. Press it. Now you should see a little white dot. This dot tells you the machine is on.

Look at the screen. It should look like this:

```
###Commodore Basic###
   15359 Bytes Free        (or 7167)
   READY.
   □
```

The word READY tells you that the PET can now do what you want it to. It is not asking you a question. The flashing white square is called the cursor and shows you your place on the screen.

Before you go on, take this check test:

Quiz #1

1. The switch for turning the PET on and off is on the front or back?
2. The white dot on the switch tells you the machine is ________?
3. The white flashing square is called the ____________.
4. The word READY is asking you if you are ready to use the PET?   yes or no

LESSON II

Calculator

The PET can do math problems. You will be asking the PET to PRINT the answer so you must use the code for that word which is ?. If you want to find the answer to 5+3 you would type it this way on the PET:

```
? 5+3
  8
READY.
□
```

The word READY tells you that the PET is ready for another problem.

There are four symbols that the PET uses:

The addition sign	+
The subtraction sign	-
The multiplication sign	*
The division sign	/

Quiz #2

1. What are the signs for: a) Multiplication ____
   b) Subtraction ____
   c) Division ____
   d) Addition ____

2. Do these problems on the PET.

   a) 5+4 b) 6x7 c) $4\overline{)\,12}$ d) 9-5

LESSON III

The Keyboard

The keyboard is a lot like a typewriter. Look at the keys. Most of the keys have a letter and a symbol. Try pressing different keys. You will see the letters printed on the screen.

Now find the SHIFT key. Hold it down and press a letter key. This time the symbol is printed on the screen.

What do you think will happen if you press the SHIFT LOCK key and then press other letter keys? Try it to see if you are right.

The SHIFT LOCK key holds the shift down until you 'unlock' it by pressing the key again. Practice typing different words and symbols.

Before you go on take this check test.

Quiz #3

1. Most of the keys have a __________ and a __________ printed on them.
2. When you press the SHIFT key and a letter key a ________ is printed on the screen.
3. What does the SHIFT LOCK key do? ________________________________

LESSON IV

Special Keys

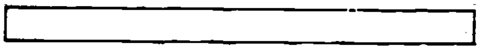
Find the long bar at the bottom of the keyboard. This is called the SPACE BAR. It prints spaces on the screen. Try printing this sentence:

> The cat ran up the tree.

Did you use the SPACE BAR between the words? If you didn't, try it again.

Find the RETURN key. This key returns the cursor to the beginning of the next line. Try it.

Quiz #4

1. What is this key? 

2. What does the return key do?

LESSON V

Loading a Program (cassette)

There are different ways to load a program from a cassette. This booklet will show two ways which will work on all PET Computers.

A.

1. Place the cassette in the recorder and rewind. (Always be sure to press the the stop key after the cassette is rewound.)

2. Type LOAD then press the return key.

   The screen will print: PRESS PLAY ON TAPE 1

3. After you press the play button the screen will print:

```
OK
SEARCHING
FOUND "name of program"
LOADING
READY.
□
```

The program is now ready to run. The PET has loaded the first program on the tape.

B.

If you have a tape with many programs on it and you want to load a program that is in the middle, here is all you do:

1. Place the cassette in the recorder and rewind.
2. Type LOAD "name of program" and press the return key.
3. The screen will print:

```
OK
SEARCHING
FOUND "name of program"
LOADING
READY.
```

Exercise: Try loading some programs using the two different ways.

LESSON VI

Running a Program

When you run a program you must answer the PET exactly. Most of the time you have to press the RETURN key after you type in your answer. Watch the screen! Some programs do it for you.

To begin a program that did not start automatically, type the word RUN and press the RETURN key. Answer the questions printed on the screen. You can type Y or N for yes and no it most programs.

If you hit the RETURN key accidentally, the program may stop. The screen will print:

```
READY
```

If you want to stop the program, press the RUN/STOP key. If you hit the RUN/STOP key the screen will print:

```
BREAK IN 'a number'
READY
```

If you accidentally break out of a program in either of these ways and wish to continue with the program you must type cont then hit return. The PET will then print a question mark and wait for you to answer the last question it asked. (Note: You must not type any other keys before doing this or the PET will not let you return. Also some programs which use graphics (Pictures) will not work correctly when you type cont) If you break out of a program and wish to start over again type run then return.

If after answering a question the PET prints:

```
? REDO FROM START
?
```

The PET probably asked a question with a number for an answer, reread the question and try a different answer.

DO'S AND DON'TS

1. If there is static electricity in the room touch your chair before touching the PET.
2. When the cassette recorder has finished loading do not leave the play button down.
3. Do not have food or drinks near the machine.
4. Do treat the PET like you treat your pet at home -- with care.
5. Do not flip the PET on and off quickly.

REVIEW QUIZ

1. The flashing white square is called:

   a) homer
   b) program
   c) cursor
   d) white square

2. When READY is printed on the screen it means:

   a) PET is asking you if you are ready
   b) nothing
   c) you told the PET you are ready
   d) PET is telling you it is ready

3. When you press the shift key and a letter key a __________ is printed.

   a) a symbol
   b) a letter
   c) a number
   d) the screen clears

4. Do these problems:

   a) 7 plus 4
   b) 8 times 9
   c) 20 divided by 4
   d) 10 take away 2

ANSWER KEY

Quiz #1

1. back
2. on
3. cursor
4. no

Quiz #2

1. a) *
   b) -
   c) /
   d) +
2. a) 9
   b) 42
   c) 3
   d) 4

Quiz #3

1. letter and symbol
2. symbol
3. holds the shift key in place

Quiz #4

1. space bar
2. moves the cursor to the beginning of the next line

Review Quiz

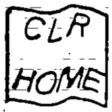
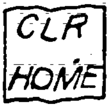
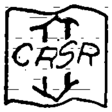
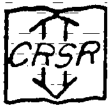
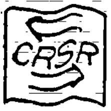
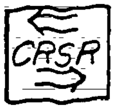
1. c
2. d
3. a
4. a) 11
   b) 72
   c) 5
   d) 8

APPENDIX A

OPTIONAL SECTION

Cursor Keys

Look at the top row of the number key pad. These 4 keys are called the CURSOR KEYS because they move the CURSOR around the screen. Without the shift key the cursor key will follow the direction on the bottom of the key. With the shift key pressed, the cursor key will follow the direction printed on the top of the key.

NO SHIFT		WITH SHIFT	
	return cursor to home		clears screen
	cursor down		cursor up
	cursor right		cursor left

Try moving the cursor around the screen.

Type your name. Now press the CLR/HOME key with the shift key. This made the screen blank. Move the cursor down, to the middle of the screen. Now press the CLR/HOME key. The cursor should have returned to the upper left hand corner of the screen. This is the HOME position.

Quiz #1

1. What do the cursor keys do? ______________________________

2. Where is the home position? ______________________________

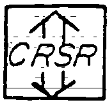
Cursor Keys (continued)

3. Fill in this chart (hint: you might not have to use all squares)

a)

= ____________

b)

+ 

= ____________

c) Cursor right = □ + □

4. Try doing the following in order.

a) move the cursor to the bottom of the screen

b) move the cursor to the middle of the screen

c) move the cursor to the lower right hand corner

d) move the cursor to the upper right hand corner

e) return the cursor to the home position

If you need help use the chart on page 1 of this section.

One More Key

This is a special key to help you correct your typing mistakes. INST stands for insert. Insert means to put in. With the shift key you can make a space in a word so you can put in a letter.

Type HUSE. You wanted to type HOUSE.

Move the cursor until it is on the U.

Press the shift and the inst del key.

You now have a space and can type in the letter O.

Press the return key.

Try correcting these words:

a) change HOSE to HORSE

b) change WERE to WHERE

c) change WITOUT to WITHOUT

The bottom part of the key--DEL--stands for delete. Delete means to take out. You do not need the shift key. Type HOURSE. You wanted to type HOUSE. Move the cursor until it is over the S. Press the INST DEL key.

The key took out the R so you now have the word house.

One More Key (continued)

Try correcting these words:

a) change CHAEIR to CHAIR

b) change CLASSRROOM to CLASSROOM

c) change HARND to HAND

Try these:

Change BLACKTOARJ to BLACKBOARD

Change FLOUIRS to FLOWERS

You can also type over wrong letters to correct your mistakes. Use the ⇐ CRSR ⇒ key to move the cursor over the incorrect letter. Now type the right letter.

Type PAPIR

Use the ⇐ CRSR ⇒ key to move the cursor over the I.

Now type an E.

Try these:

change taple to table

change store to stone

change flime to flame

Quiz #2

1. What does the INST key do? ______________________________

2. What does the DEL key do? ______________________________

3. The INST DEL key needs the shift key? (choose one)

4. What can the space bar do?

5. Correct these words:

change HORSE to HOUSE
change CAERPETS to CARPETS

change RUG to ROUGE
change BLENKATS to BLANKETS

OPTIONAL REVIEW QUIZ

1. When you press the CLR HOME key

   a) the screen clears
   b) the cursor return to the home position

2. The home position is:

   a) upper right hand corner
   b) lower left hand corner
   c) upper left hand corner
   d) lower right hand corner

3. Change HORSE to HOUSE.

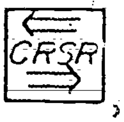
4. Change: The cat ran up the tree. to
   The cat ran down the tree.

5. Change COMPTER to COMPUTER.

ANSWER KEY FOR OPTIONAL SECTION

Quiz #1

1. moves the cursor
2. upper left hand corner
3. a) moves cursor to home position
   b) moves cursor up
   c)



Quiz #2

1. creates a space
2. takes out a space, erases letters
3. INST
4. erases letters

Review Quiz

1. 6
2. c