Design and Implementation of Computer Assisted Instruction in Money Concepts for Elementary Students

Jack H. Bender

Western Michigan University

Follow this and additional works at: https://scholarworks.wmich.edu/masters_theses

Part of the Curriculum and Instruction Commons

Recommended Citation
https://scholarworks.wmich.edu/masters_theses/1633

This Masters Thesis-Open Access is brought to you for free and open access by the Graduate College at ScholarWorks at WMU. It has been accepted for inclusion in Master’s Theses by an authorized administrator of ScholarWorks at WMU. For more information, please contact maira.bundza@wmich.edu.
DESIGN AND IMPLEMENTATION OF COMPUTER ASSISTED INSTRUCTION IN MONEY CONCEPTS FOR ELEMENTARY STUDENTS

by

Jack H. Bender

A Project
Submitted to the
Faculty of The Graduate College
in partial fulfillment of the
requirements for the
Degree of Specialist in Education
Department of Educational Leadership

Western Michigan University
Kalamazoo, Michigan
April 1982
DESIGN AND IMPLEMENTATION OF COMPUTER ASSISTED INSTRUCTION IN MONEY CONCEPTS FOR ELEMENTARY STUDENTS

Jack H. Bender, Ed.S.
Western Michigan University, 1982

The internship with Hartley Courseware, Inc. culminated with the creation of nine money concept programs. The programs were designed for a 48K Apple II microcomputer (trademark of Apple Computer, Inc.) with Applesoft in ROM and one disk drive. Text size was expanded through use of the Minnesota Educational Computing Consortium's high resolution text subroutines.

A placement program tests students' abilities and recommends the appropriate starting point. Six programs introduce information on coin values and numerous money concepts. The later programs develop sequences of actions in the purchasing process. The practice test and test programs progress through the complete purchase procedure. Student scores on all programs are kept on the diskette for teacher use.
ACKNOWLEDGEMENTS

My internship was an exciting and successful learning experience.

I wish to express my appreciation to Tim Hartley of Hartley Courseware, who gave unselfishly of his time and knowledge in his capacity as field supervisor.

My sincere appreciation to Dr. James R. Sanders, Advisor, and Dr. Kenneth E. Dickie, committee member. Your guidance, support, and example were important elements in this venture.

My thanks to Dr. Harold W. Boles, Dr. David J. Hamilton, Dr. John E. Herman, and Dr. Kenneth L. Williams for allowing me to meet my personal educational goals.

To Cindy, Jenny, and Joe, who gave special meaning to the words patience and understanding, all my love.

Jack H. Bender
INFORMATION TO USERS

This was produced from a copy of a document sent to us for microfilming. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help you understand markings or notations which may appear on this reproduction.

1. The sign or “target” for pages apparently lacking from the document photographed is “Missing Page(s)”. If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure you of complete continuity.

2. When an image on the film is obliterated with a round black mark it is an indication that the film inspector noticed either blurred copy because of movement during exposure, or duplicate copy. Unless we meant to delete copyrighted materials that should not have been filmed, you will find a good image of the page in the adjacent frame. If copyrighted materials were deleted you will find a target note listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed the photographer has followed a definite method in “sectioning” the material. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.

4. For any illustrations that cannot be reproduced satisfactorily by xerography, photographic prints can be purchased at additional cost and tipped into your xerographic copy. Requests can be made to our Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases we have filmed the best available copy.

University Microfilms International
300 N. ZEEB RD., ANN ARBOR, MI 48106

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
BENDER, JACK HUGH
DESIGN AND IMPLEMENTATION OF COMPUTER
ASSISTED INSTRUCTION IN MONEY CONCEPTS FOR
ELEMENTARY STUDENTS.

WESTERN MICHIGAN UNIVERSITY, ED.S., 1982
PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark √.

1. Glossy photographs or pages _____
2. Colored illustrations, paper or print _____
3. Photographs with dark background _____
4. Illustrations are poor copy _____
5. Pages with black marks, not original copy _____
6. Print shows through as there is text on both sides of page _____
7. Indistinct, broken or small print on several pages _____
8. Print exceeds margin requirements _____
9. Tightly bound copy with print lost in spine _____
10. Computer printout pages with indistinct print _____
11. Page(s) ___________ lacking when material received, and not available from school or author.
12. Page(s) ___________ seem to be missing in numbering only as text follows.
13. Two pages numbered ___________. Text follows.
14. Curling and wrinkled pages _____
15. Other ____________________________________________________________________

University Microfilms International
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>ACKNOWLEDGEMENTS</th>
<th>ii</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CHAPTER</strong></td>
<td></td>
</tr>
<tr>
<td>I INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>Background</td>
<td>1</td>
</tr>
<tr>
<td>Site and Project Organization</td>
<td>2</td>
</tr>
<tr>
<td>II THE PRODUCT TO BE DEVELOPED</td>
<td>3</td>
</tr>
<tr>
<td>Objectives</td>
<td>3</td>
</tr>
<tr>
<td>Literature Review</td>
<td>3</td>
</tr>
<tr>
<td>Design</td>
<td>4</td>
</tr>
<tr>
<td>III EXPERIENCES</td>
<td>10</td>
</tr>
<tr>
<td>Instructional</td>
<td>10</td>
</tr>
<tr>
<td>Programming</td>
<td>16</td>
</tr>
<tr>
<td>Interpersonal Relationships</td>
<td>22</td>
</tr>
<tr>
<td>IV ACHIEVEMENTS</td>
<td>26</td>
</tr>
<tr>
<td>Description of the Product</td>
<td>26</td>
</tr>
<tr>
<td>Knowledge and Skills Gained</td>
<td>33</td>
</tr>
<tr>
<td>V CONCLUSIONS AND RECOMMENDATIONS</td>
<td>36</td>
</tr>
<tr>
<td>APPENDICES</td>
<td></td>
</tr>
<tr>
<td>A PROSPECTUS FOR INTERNSHIP</td>
<td>38</td>
</tr>
<tr>
<td>B COURSE OFFERINGS OF HARTLEY COURSEWARE</td>
<td>44</td>
</tr>
<tr>
<td>C AVAILABLE SOFTWARE</td>
<td>46</td>
</tr>
<tr>
<td>D DEFINITION OF TERMS</td>
<td>50</td>
</tr>
</tbody>
</table>
CHAPTER I

Introduction

Background

As microcomputers come into use as tutors and record keepers in schools, there is a need for qualified line and supervisory personnel. Programming, software selection, scheduling of users, curriculum planning, information handling, literacy of staff and students, and hardware selection are only a few of the various activities that are, or soon will be, facing individual schools. Existing publishers and new software firms are seeking qualified programmers, consultants, and supervisors.

Supervisors need an adequate exposure to both educational and programming activities. A sense of what has been done, and what is feasible is important. A sense of time requirements is useful and the ability to coordinate individual activities into a team effort is vital.

The internship offered experiences in planning and programming instruction. This experience provided the conceptual and technical skills needed to assume a leadership role in education, which would aid others in the educational community.
Site and Project Organization

Hartley Courseware, Grand Rapids, Michigan, is devoted to the creation and marketing of educational Apple II* microcomputer software for grades K-8. The firm offers 29 packages and a computer controlling device for cassette tape recorders. Tim Hartley is chief programmer. Jane Hartley and Rosie Bogo are off-site educational consultants. All travel throughout the United States to speak on educational software topics. The Hartleys have been featured in Softalk, a software magazine, for their work in the field of educational programming.

The project was selected by the field supervisor, Tim Hartley, and centered on money concepts for grades 3-6. It was decided that a set of tutorials be developed, which would introduce coin values, quantitative concepts, and culminate with store-like money transactions.

*Apple II is a trademark of Apple Computer, Inc.
CHAPTER II

The Product to be Developed

Objectives

The goal of the product is to teach money concepts. The terminal objective is the successful purchase transactions by students when given an item, price, and an amount of money on hand. A pretest is expected to place students within a series of programs according to their ability. A test will be administered at the completion of the money concept programs. Student scores are to be recorded on the diskette.

Literature Review

A literature review for the need of this product was conducted. As indicated on page 11, five known products existed, but appeared less comprehensive in scope than the product to be developed.

Existing programs presented either store transactions but not tutorials, or drill and practice but not store transactions. Three packages of the five were for computers other than the Apple II. Programs are not transferable between brands of computers without modification.
Design

Educational planning included the following outline:

Referent situation: Students need factual information about money. Students also need skills used to make purchases at stores. The enclosed programs are designed to help students gain money usage skills. These are important skills for they are needed by everyone for everyday living.

Problem definition: Young people are unable to absorb information for many reasons. Among these are:

1) slow learning rate
2) mismatch of teaching style with learning style
3) low ability
4) absence from school
5) exclusion or cursory treatment of subject matter
6) lack of practice or feedback
7) short attention span

As a result of the presence of any of the above reasons, understanding may not be reached. This deficiency in understanding the concepts and principles related to money exists in varying degrees in young people. How can this situation be remedied?

Goal of program: Students will understand the fundamental concepts and principles related to money and be able to use small bills and coins in common money transactions.

Terminal objective: Given a price tag and an amount of money, the student will (1) indicate whether he has sufficient money for the
purchase, (2) present the lowest sufficient amount of money to buy the item, (3) upon receiving change, the student will indicate whether the change is correct. The student should successfully complete the transaction five out of six times.

**Audience characteristics:** Due to national exposure, audience characteristics vary greatly. The following student entry skills are assumed:

1) Students possess grade 3 reading level.
2) Students are able to count by 5's to 100.
3) Students are able to add, carrying into the hundred's place.
4) Students are able to subtract, borrowing from the hundred's place.

From field testing a sight word list should be developed which outlines individual words not found in third grade student vocabulary.

**Constraints:** Although the cassette control device (CCD) would be advantageous to auditory learners, it was anticipated that its additional cost might discourage use of this program. Therefore, only visual stimulus is provided. The program set requires an Apple II, disk, Applesoft, in ROM and 48K. Due to TV monitor size differences and the resolution of the graphics, coins will be presented in sizes relative to each other only. Because of the lack of control of the program application and due to the great variability of the audience, the design has not revolved around more specific audience traits than those listed in the section above.
Resources: A money card will be available to the student. The card consists of actual size photos of coins and is used for coin identification and concept development. Programs also use TV graphic representations of the coins.

Environment: It is recommended that one child use the program at a time and that an appropriate study environment be provided.

Task description: The task can be described as having a fixed sequence of actions. The actions are as follows:

1) Count how much money is on hand.
2) Select items in store.
3) Determine whether the amount on hand is adequate for a purchase.
4) Take items to cashier.
5) Notice what cashier says (cost).
6) Calculate lowest sufficient amount to be paid.
7) Select money from that which is on hand.
8) Notice register and cashier (change).
9) Count change.
10) Compare expected change and actual change.

Task analysis: Because of individual differences in experiences with money, a multiple entry learning program should be adopted. A placement test will be utilized. The word 'rule' below may be closely related to 'principle.'
<table>
<thead>
<tr>
<th>Program Number</th>
<th>Subject</th>
<th>Learning Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>coins</td>
<td>concept</td>
</tr>
<tr>
<td>1</td>
<td>dollar bill</td>
<td>concept</td>
</tr>
<tr>
<td>1</td>
<td>dollar bill and coin identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>1</td>
<td>dollar sign</td>
<td>concept</td>
</tr>
<tr>
<td>1</td>
<td>bill</td>
<td>concept</td>
</tr>
<tr>
<td>2</td>
<td>penny</td>
<td>concept</td>
</tr>
<tr>
<td>2</td>
<td>penny identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>2</td>
<td>cents</td>
<td>concept</td>
</tr>
<tr>
<td>2</td>
<td>cent sign</td>
<td>concept</td>
</tr>
<tr>
<td>3</td>
<td>nickel</td>
<td>concept</td>
</tr>
<tr>
<td>3</td>
<td>nickel identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>3</td>
<td>change</td>
<td>rule</td>
</tr>
<tr>
<td>4</td>
<td>dime</td>
<td>concept</td>
</tr>
<tr>
<td>4</td>
<td>dime identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>4</td>
<td>quarter</td>
<td>concept</td>
</tr>
<tr>
<td>4</td>
<td>quarter identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>2,3,4</td>
<td>equivalents of coins</td>
<td>concept, rule</td>
</tr>
<tr>
<td>2,3,4</td>
<td>value of coins</td>
<td>rule</td>
</tr>
<tr>
<td>5</td>
<td>not enough money</td>
<td>concept, rule</td>
</tr>
<tr>
<td>5</td>
<td>lowest sufficient amount</td>
<td>concept, rule</td>
</tr>
<tr>
<td>5</td>
<td>adding coins until lowest sufficient amount is reached</td>
<td>rule</td>
</tr>
<tr>
<td>6</td>
<td>half dollar</td>
<td>concept</td>
</tr>
<tr>
<td>6</td>
<td>half dollar identification</td>
<td>discrimination</td>
</tr>
<tr>
<td>6</td>
<td>decimal point</td>
<td>rule</td>
</tr>
</tbody>
</table>
one dollar and 23 cents = $1.23 rule
sixty-seven cents = $.67 rule

The following frames are to be used in checking for accomplishment of the enabling objectives. Informational frames are ignored in student response record keeping, so both forms are listed.

<table>
<thead>
<tr>
<th>Frame</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>(bill)</td>
<td>5</td>
</tr>
<tr>
<td>(coin)</td>
<td>6</td>
</tr>
<tr>
<td>(value)</td>
<td>14, 17</td>
</tr>
<tr>
<td>(dollar sign)</td>
<td>20</td>
</tr>
<tr>
<td>(penny)</td>
<td>22, 27</td>
</tr>
<tr>
<td>(cents)</td>
<td>23</td>
</tr>
<tr>
<td>(cent sign)</td>
<td>24, 25</td>
</tr>
<tr>
<td>(nickel)</td>
<td>21, 24, 27</td>
</tr>
<tr>
<td>(change)</td>
<td>14, 18, 19</td>
</tr>
<tr>
<td>(equivalent values)</td>
<td>20, 21, 22, 23</td>
</tr>
<tr>
<td>(dime)</td>
<td>8</td>
</tr>
<tr>
<td>(quarter)</td>
<td>15, 16</td>
</tr>
<tr>
<td>(equivalents)</td>
<td>9, 18, 20, 21</td>
</tr>
<tr>
<td>(counting by 5's)</td>
<td>27</td>
</tr>
<tr>
<td>(enough)</td>
<td>9</td>
</tr>
<tr>
<td>(exact amount)</td>
<td>11</td>
</tr>
<tr>
<td>(lowest sufficient amt.)</td>
<td>22, 23, 24</td>
</tr>
<tr>
<td>(not enough money)</td>
<td>4, 10</td>
</tr>
<tr>
<td>(counting by 5's)</td>
<td>15, 16</td>
</tr>
<tr>
<td>(half dollar)</td>
<td>5, 8</td>
</tr>
<tr>
<td>(---)</td>
<td>15, 16</td>
</tr>
<tr>
<td>(3 forms of ---)</td>
<td>17, 18, 20, 22, 25, 26, 27</td>
</tr>
<tr>
<td>(equivalents)</td>
<td>7</td>
</tr>
<tr>
<td>(adding)</td>
<td>32, 33</td>
</tr>
<tr>
<td>(subtracting)</td>
<td>34, 35</td>
</tr>
</tbody>
</table>

Measurement: For measuring competence in the terminal objective, a set of six visits to a store will be presented. Five visits will include the following frames:
Stimulus: Money printed graphically.
Response: How much is present?

Stimulus: Price of item and money on hand.
Response: Affordable or not?

Stimulus: Money printed graphically and item price.
Response: Lowest sufficient amount of money necessary?

Stimulus: Item price and money tendered.
Response: Amount change should be?

Stimulus: Change drawn graphically and numerically.
Response: Change correct or not?

Each skill will be scored separately. An incorrect response will not jeopardize performance on other frames. Later frames of the same transaction will always give correct information, instead of an incorrect response. The student's score will reflect ability in:

1) adding coin values,
2) knowing if money on hand is adequate for a purchase,
3) figuring lowest sufficient amount,
4) subtraction,
5) comparing change received with expected change.

The student will have insufficient funds during one visit and the full sequence will be aborted after two frames. Incorrect change will be given once during the six visits.
CHAPTER III

Experiences

Under the direction and supervision of Tim Hartley, Hartley Courseware, the bulk of the internship was centered around the educational planning and implementation stages of instructional development. Product evaluation procedures were not undertaken. The development of nine programs was completed. Tim Hartley acted as liaison with Rosie Bogo and Jane Hartley, off-site educational consultants.

Instructional

The initial sessions with the supervisor dealt with defining the product to be produced. The programs were expected to use graphics for text and coin images, provide feedback, be sequenced, be user proof, and keep records of student scores. Math Concepts, a new program of the firm, was offered as an example and study guide for the money programs.

The firm anticipated a need for the program. A number of teachers had asked the firm to develop a program which would teach coin values and coin equivalents. The firm was not aware of tutorials in the money concept area. The consultants suggested a third grade level with expectations that remedial students in grades 4-6 would also use the program.
After the initial session, a literature review to determine need followed. Various issues of *Creative Computing*, *The Computing Teacher*, and the *MACUL Journal* were used. The MicroMedia Software catalog, a clearing house for a number of firms, yielded the largest number of programs. Five programs related to money concepts, by five different sources, were found (see Appendix C). Each program appeared useful, but not nearly as comprehensive as the new product was expected to be.

A literature review of written materials, teaching guides, and learning modules dealing with money followed. The basic objective was to determine how money concepts were being taught and the instructional activities which would be familiar to students. Sources aimed at other than third graders were also used to better understand what activities students may already have experienced. Instructional strategies were noted. A "feel" for traditional methods was obtained.

Techniques and activities included the following:

1) Use of glass piggy banks to teach the concept of addition.
2) Mark a frame which identifies a coin.
3) Mark a frame which will buy this much (coin pictures or names used as stimulus).
4) Add coin values to get total amount.
5) Pick which combinations of given coins will buy an object.
6) Act as cashier and store patron.
7) Pick cafeteria items and pay for them.
8) Figure the change from given price and amount paid.
9) Express money in dollar and decimal point form.

Following the literature review, the educational plan was developed. Since the firm did not offer an instructional development model, the model outlined by Davis\(^1\) was utilized.

In the next planning session with the supervisor, the literature reviews were reported and the terminal objective set. It was jointly agreed that the terminal objective should be based upon a purchase transaction at a store. Such a referent situation would provide motivation for the student and would certainly encourage transfer of skills learned to life situations.

Using the Davis model, the modules were developed. A discussion with the supervisor uncovered a philosophy which carefully avoided use of auditory reinforcement and gaming strategies. Feedback using sound seems fine at first, but is disruptive in a classroom and tedious after long exposure. Learning which is highly interactive and successful does not require additional forms of motivation such as gaming to hold student interest.

Prior to this session, the intern had brainstormed a number of motivational or gaming strategies:

1) "Crossing into Canada" involved money exchanging.

2) "Nickelodeon" dealt with feeding coins to a discriminating coin slot.

3) "Trade" tested making change for a dollar bill.

4) "Hey Buddy Gotta Dime?" uses a panhandler to ask for specific money.

5) "Kissing Booth" has Snoopy charging for kisses.

6) "Fee Fi Fo Fum" has a giant finger counting money.

7) "Feed the Meter" dealt with time and coins needed.

Many of the firm's programs rely on a computer controlled tape recorder to provide auditory stimulus, thereby reinforcing the visual stimulus on the screen. The supervisor expected that clients without the cassette tape controlling device would be discouraged from buying the product due to the additional cost.

Random generation of learning material was discounted because of the tutorial nature of the material and the expected need for sequential presentation.

The use of graphics to display likenesses of coins was encouraged. After a more thorough consideration of the types of learning involved, it became evident that realistic coin representations would be necessary. Early in the program students would see or handle coins to aid in concept formation of coin identity and value. It was jointly agreed that a sheet of life size color photos would be included showing coins on both sides. The program would direct the student to the sheet for detailed study of each coin.

The intern was allowed great freedom in decision-making. The following decisions, in particular, were made.

Since linear programs were to be used, there existed a problem of student entry. Two choices seemed possible. Pretest items
could appear at the beginning of each program or a comprehensive pretest could be given initially. A comprehensive version was chosen so students would not have to load six programs to determine their starting point. A criterion was established for passing each part of the placement test.

Presentation of coin identification and values seemed to be the starting point for instruction. The five values, five names, and ten sides of the coins not only seemed too much for one program but seemed confusing. It was anticipated that "interference" in the learning would result. The coins were spread throughout the programs to avoid interference.

User considerations included input and feedback decisions. It was decided that a multiple choice answer format would be used. Because of the variety of forms answers could take and the lack of strong typing ability, input errors might result. However, to test total recall, some frames would accept unprompted responses (fill in the blank).

Two common modes exist for feedback on incorrect frames. One mode is to indicate an incorrect response and supply the answer. The other mode is to indicate an incorrect response and require another response on the frame. It was decided to provide the answer after an incorrect response. There would be many opportunities for practice on each concept, therefore providing the answer was viewed as providing another example on which to base future behavior.
After the programs were outlined, a summary seemed needed. The summary took the form of a practice test. It was expected that taking the test would be useful practice before the final test, and would sensitize the student to the full purchase procedure and to the test itself.

Some initial evaluative procedures could have been accomplished using pencil and paper drafts of the frames. It was decided, however, that the computer offered such good record keeping abilities as to encourage initial evaluation using the computer itself.

During field testing the consultants normally rate the reading level after the program has been used by many students. The intern wished to document the program's reading level using a recognized instrument. Costello² provided an overview of possible readability formulas. The Fry Graph method was used. A 100 word sample was taken from programs 1, 3, and 5. For every 100 words the average number of syllables was 126 and the average number of sentences was 18.8. Readability using Fry's graph was at the second grade level. Fry's method cannot be relied upon fully. Since averaging takes place, there may be words which some students would be unable to read. Knowing the method determined a second grade reading level for the program, confidence was increased for success with third grade students.


Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
During the internship, the firm was not developing a major tutorial type program. Since the money concept programs were tutorial, a model was not available for study and/or emulation. On the other hand, the freedom given to make decisions and mistakes was very valuable.

Programming

The experiences related to software considerations seemed to be the most illuminating. Prior to the internship, nearly all the intern's experiences had been with non-interactive, non-graphic applications. The products developed were interactive and relied heavily on graphics.

A literature review was not done in this area. However, much information was gained through reading brief sections in the hardware manufacturers' manuals and studying the programs of Hartley Courseware already on the market. The study of program lines brought the intern to two conclusions: (1) code can be very powerful, yet terse, if constructed effectively, and (2) a programmer must know intimately the hardware capabilities of a given machine to write such code.

The first program lines to be studied involved the Minnesota Educational Computing Consortium's hi-resolution graphics subroutines. The supervisor had just received permission to utilize the subroutines in the firm's software. The hardware prints only upper-case letters using a 5 x 7 dot matrix. There can be a maximum of 40 characters on a line with 24 lines.
subroutines afforded lower case letters by using the graphics' capabilities. Included in the MECC package were three shape tables which facilitated drawing text in three different sizes. It was decided to utilize the medium size letters which utilize a 9 x 11 dot matrix for lower case letters. Maximum lower case letters per line is 30 and maximum upper case letters is 25. The medium size was chosen to allow a good deal of text per frame yet improve over the hardware in readability and lower case. It is likely that the use of the subroutines saved over two hundred hours in development because of the tedium involved in creating shapes and shape tables.

The decision to use the hi-resolution screen automatically limited the availability of programmable memory, reducing it by approximately 35% or more. Through use of the second hi-resolution graphics page to print text, further memory loss was saved. The graphics area of memory can be used for either storage of program lines or graphics, not both. If the program becomes long enough, the statements begin to be stored in the graphics' area. When using graphics, the hardware automatically clears the graphics' memory area. This in effect clears the TV screen to black and allows the drawing of text or pictures. By clearing memory, the program lines would be erased and the program rendered inoperable. With the use of Hi-Res graphics, it became obvious that the program needed to be written compactly.

The use of the LOMEM command was studied to aid in efficient memory utilization. Through the use of this command, program
variables can be stored in free memory areas other than those used for graphics and program storage.

Coin and bill values and identification were learning objectives. The graphics' capabilities afforded an adequate amount of resolution to draw these figures on the screen. Sixteen shapes were created on graph paper. Graph paper with very small divisions was used and a dot was placed in each appropriate square. A starting place on the figure was selected and vector arrows were drawn. To draw a figure, the computer is given one of two types of directions. Either the computer plots a dot and moves left, right, up, or down, or the computer does not plot a dot and moves in any direction. Directions are given one at a time as progress is made through the figure. The vector arrows are then interpreted into binary numbers and then into hexadecimal numbers. The hexadecimal numbers are entered into memory from the keyboard and then saved to disk.

The task of creating the shapes was very tedious. However, an appreciation and understanding of the time involved with the use of graphics was gained. The intern became much faster at creation of the shapes as experience was gained. Coins having solid edges were created and were found lacking in visual preciseness. The milled edge of the coins was simulated by alternating blank spaces and dots. See Appendix E for samples of shapes. Due to the amount of detailed work (dots, arrows, binary, hexadecimal), correction of small errors was necessary.

The cent sign does not exist on the keyboard nor in the ASCII
set of character codes, so the cent character shape was created to match the text size.

It was suggested that coins and bills be drawn to scale to aid students' concept learning. It was suggested that a head and tail version be created. One version would supply the amount of the coin in the center and the other would supply the coin name. The use of each would be determined by learner activities. To determine the scale, a 5 x 7 text size for coin names was selected. Coin name lengths helped narrow down the choices until a coin diameter of 1 mm = 1 dot was chosen. Dollar bills were found to be too large according to this formula and were reduced to a "tasteful" size. Coins are relative in size to each other.

The program statements studied next were related to user input. Many errors can result from user misunderstanding or misuse. The routine used by the firm eliminated these errors which normally stop program execution. Much was learned from this study. The main idea is to analyze each character of user input as it is received and to adjust conditions if needed. This is contrasted with accepting user "garbage" and halting due to an error.

When discussing the unavailability of the cent sign on the keyboard, the supervisor indicated three useful characters which could be obtained that are not on the keyboard. By assigning each ASCII character code to a different string variable during a program, these characters are printable. The underscore is sadly missing from the keyboard but may be obtained in this manner.

A critical comment about the Apple II is that the reset key
is placed too close to the return key. The reset key is used to halt a program execution. This is not desirable. The return key is frequently used. Many users press the reset key instead of the return key. The supervisor showed how to avoid this problem by removing the reset key and wrapping a rubber band around a column beneath the key top. The reset key is then placed back on the column. It now takes greater pressure to halt program control than is usually applied to the return key.

The next statements studied were those related to the student score text file. A student's incorrect responses were retained in memory along with the correct response. The student's name, program name, number of questions completed, and number correct were also retained. Upon completion or premature termination of the program, this data is saved to disk. The statements provided many useful ideas.

It was at this point that a major programming decision was made. It had not been decided whether programs created during the internship would be completely original or based on the firm's current offerings. Because of limited time at the firm, the need for the supervisor to understand and update any work in the future, and the wasteful duplication of effort, it was jointly decided to use appropriate subroutines of the firm's.

The normal procedure for loading a program is to display the names of the programs and then have the user type the name of the desired program. Other statements were studied which showed how to handle the mistype of program names. These statements are
necessary or a "file not found" error will result. The intern suggested a menu in which the user picks a program by number, eliminating misspelling. The menu will only accept legitimate numbers.

The supervisor suggested that the use of a program line editor would speed up the process of altering program lines. The program line editor is loaded into memory, and protected from the program being fixed. The line editor's capabilities are more powerful and convenient than the resident editor's capabilities, thereby making changes more rapidly.

A general session with the supervisor uncovered a number of techniques used in protecting copyrighted programs. It remains common practice to copy copyrighted disks and trade or give to others thus avoiding the purchase price. Four procedures are used to help protect programs from being copied. Disks can be made uncopiable by changing a number of memory locations to certain values. The DOS located in RAM can be altered to override listing of the program statements to a printer or TV screen. Lines above 63999 are not erasable by ordinary means. These lines can be placed in memory at the end of a program leaving a permanent copyright message. The DOS catalog of programs can be altered by unusual means to place copyright information on the catalog. When a user wishes to obtain a list of programs stored on a diskette, the copyright information will be displayed. These four processes were new to the intern. Also included in this session was a procedure to load and execute programs at other than the default
place in memory. A program must be created to place new values in memory designating where the program should start and then load the intended program to be run.

As stated earlier, the experiences gained in the programming portion were very worthwhile. Good programming skills will allow the ideas of the instructional designer to be realized, instead of becoming subservient to an ill-conceived program.

Interpersonal Relationships

The personnel at Hartley Courseware, Inc. engaged in separate and shared activities. The consultant was responsible for all process and content work. The programmer was responsible for computer code which conveyed the intended process and content.

The programmer handled the protection measures already mentioned, constructed routines, projected memory usage, diskette space, loading time for text files, and performed other related tasks.

The consultant developed the content, determined whether a random, linear, or branching approach would be most suitable, developed delivery strategies, determined sequence, and performed other related tasks.

A number of important considerations were discussed jointly. If the program operating instructions were written by the programmer, the consultant would check for reading level. The presentation of stimuli was important. Size of the text font, means of student response, and feedback methods were coordinated. Much of the coordination was done by phone as the consultants were off-site much
of the time.

The supervisor attended many conferences during the internship where a booth was set up to display the firm's products. Many program ideas came from conference attendees. Some attendees would indicate that they had already purchased software from the firm and would then proceed to suggest follow-up programs or new features that might be added to the current software. Other attendees would simply suggest programs in subject areas that had yet to be covered by reputable firms. Attempts were made to see how such new programs would satisfy needs. A prioritized list is kept on programs under consideration for development. The number of requests for a given program are recorded. If demand is great enough, programming feasibility and marketability decisions are made.

Besides conference contacts with teachers, the firm receives a number of requests for programs by mail. Rosie Bogo, consultant, travels extensively each year as a consultant and becomes aware of other needs in the field.

Although no product evaluation was conducted, the intern uncovered the following procedures which relate to observing the product in the hands of students. Huntington Woods Elementary, Wyoming Michigan, and Ionia Elementary, Ionia, Michigan, are used as field testing sites. Rather informal procedures are used with students to obtain data on the usability and effectiveness of the program. Students are encouraged to request assistance so that problem areas in the operation and learning phases can be
pinpointed. No program is expected to be marketed until some revision occurs. Two to three weeks are adequate to uncover most weaknesses. Versions are revised until at least a week's use with new students brings little or no questions about operation or content.

Judgment as to merit and worth are made from records on the diskette and personal observation. The position of the firm is not to set a minimum performance criteria. Due to a program's broad application on a national level and the importance of the individual teacher, it has been the firm's policy not to establish minimum acceptable performance criteria.

The intern cannot say that evaluation measures were not adequate, particularly since they were not observed. However, at a minimum, the purpose of instruction should be to change behavior to an acceptable level. The level must be specified. Without some confidence as to the ability of the program to change behavior, clients can never be sure if the program is effective.

Time spent in the major activities was proportioned in the following manner:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receiving advice</td>
<td>5%</td>
</tr>
<tr>
<td>Review of literature</td>
<td>4%</td>
</tr>
<tr>
<td>Writing the educational outline</td>
<td>5%</td>
</tr>
<tr>
<td>Writing the frames</td>
<td>11%</td>
</tr>
<tr>
<td>Developing the graphics</td>
<td>24%</td>
</tr>
<tr>
<td>Studying programs for continuity</td>
<td>4%</td>
</tr>
<tr>
<td>Developing and keying in programs</td>
<td>44%</td>
</tr>
</tbody>
</table>
Debugging, formatting, documenting 3%
CHAPTER IV

Achievements

Description of the Product

Described herein are the features and general organization of the product. Students have access to eleven programs. Teachers can have access to three additional programs used in educational management.

When the student turns the computer on, the first program is automatically loaded. By convention, the first program is called 'Hello.' In this situation, 'Hello' provides a money concept title, copyright information, pauses for the reader, and then automatically loads the second program.

The second program is called 'Student Menu.' A menu is displayed on the screen which shows the program names available to the student. By typing a single letter any one of the nine learning programs can be loaded automatically. Typing errors are handled so only legitimate entries are accepted and acted upon by the program.

The first program to be selected by students is the placement exercise. This program requests the student's name and proceeds to explain how the program works. There are three types of frames to be encountered. Informational frames require pressing the space bar to proceed to the next frame. Multiple choice frames
require use of the arrow keys to manipulate a hand on the screen which can point to the correct answer. The final type of frame requires typing of the answer. This indoctrination not only covers the placement exercise, but serves as a model for all programs since the response features remain the same.

The placement exercise has four to six frames from each of the six tutorial programs. If the student does not miss more than one frame per section, he continues to the end of the program. If the student misses more than one frame in a section, the program branches to the end. At the end the student is given the program name on which to start. The student's name and starting point are then placed on the diskette for use by the teacher. The placement program then loads the menu back in for convenient selection of the student programs.

The six tutorial programs have from 20 to 35 frames. All utilize informational frames and the two methods of responding as described earlier. If the student's response to the frame is correct, the screen is cleared and a large 'CORRECT' appears. If the student's response is incorrect, the portion displaying the responses is cleared, leaving the stimulus. 'Answer is' followed by the correct answer is printed below the stimulus. The frame will remain displayed until the space bar is pressed.

At the end of each tutorial the student's name, program name, frames attempted and frames correct are displayed. The incorrect frame numbers are displayed along with the student's response and the correct response. All of this information is then saved on
diskette for teacher use.

The student may exit a program before completion by pressing the escape key. Typing error handling routines prevent the programs from stopping or hanging.

Program one contains 20 frames. Concept formation for bills, coins, value, and the dollar sign are introduced. After informational frames are presented, typical frames which provide practice and feedback include:

Frame 5    Point to the bill.
Frame 6    Point to the coin.
Frame 14   Which bill buys more?
Frame 19   Point to the dollar sign.

Program two contains 27 frames. Concepts of penny, cents, and cent sign are introduced.

Frame 3    We use a sign for 'cent.'
           It is the first letter of 'cent' and a line.
        ć

Frame 11   A dollar can be divided into 100 small parts. Each part is called a ________.

Frame 17   The penny is the only ________ coin.
           brown, silver, grey, black

Frame 21   The coin with a value of one cent is a ________.

Frame 23   Cent means ________.
           nickel, dollar bill, part of a dollar

A chart is used to develop the concept of individual coins. The chart has not gone into production as yet. The chart layout has nine boxes which are labeled A-I. Life-size, color photos of
coins are used to help show size and color variations. The chart will display head and tail versions. Some boxes are used for information such as, "'B' on the chart shows a nickel." Some boxes are used to test information learned such as, "The coin at 'E' is a _____."

Program three develops the concepts of nickel, change, and equivalent values (5 pennies equal 1 nickel). Program three has 27 frames.

Frame 1  'B' on the chart shows a nickel.  The color of the nickel is _______.

Frame 5  3 pennies are worth 3 cents.  How many are worth 5 cents?

Frame 7  ________ pennies equal a nickel.

Frame 12 Money left over is 'change'.  Paying 5¢ for a 3¢ item, the change is ________.

Frame 19 An item is 5 cents.  You have a nickel.  Is there change?

Frame 27 The coin at 'F' is a _______.

In 27 frames, program four introduces the dime and quarter and expands the coin equivalency drill. Counting by 5's as a means of adding coins is introduced.

Frame 6 Paying a dime for a 5 cent item, your change is ________.

Frame 11 25 pennies are heavy in your pocket!  You can carry a 'quarter' instead.  A quarter is worth ________.

Frame 22 An item is 30 cents.  You pay a quarter and a ________.  

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Frame 25  Counting by 5's you can add the nickels you have.

Program 5 uses 24 frames to introduce the rules of exact amount, lowest sufficient amount, and not enough money. Counting by 5's to figure lowest sufficient amount to tender is also included.

Frame 2  Cost is 25 cents. You have ... (coins printed graphically here). Can you buy the item?

Frame 10  Item is 37 cents. You have ... (coins printed graphically here). You have enough? not enough? exact change?

Frame 12  What number is missing? 5 10 ____ 20 25

Frame 15  Counting by 5's the number larger than 26 is ____.

Frame 17  When you have enough money but not exact change, count by 5's. Give coins equal to the 5's number.

Program six covers the half dollar, decimal point, and dollar sign, and further equivalents. The three forms of the dollar and decimal point are covered such as even dollars ($3.00), under a dollar ($.67), and dollars and cents ($1.45). Adding and subtracting money in this form is covered.

Frame 14  There are _______ numbers to the right of the decimal.

Frame 15  $3.25 means 3 dollars and 25 cents. The numbers to the right of the decimal are _______. coins, cents, change, dollars

Frame 24  Type in six dollars and 21 cents. Include the decimal.
Frame 28  You can still add.
Look at these carefully.

$1.00  $.20  $1.00  $.20  $2.00  $.40

Frame 34  Use subtraction.

$1.00  .30

The practice test combines previously learned materials with a series of activities which parallel the terminal objective. Students receiving a maximum score on the placement program are sent to this program prior to taking the test. As with all programs, a linear approach is used. Conditions much like a transaction at a store are provided. When conditions are such that the student does not have enough money, the program does skip out of the purchase sequence and into the next viable transaction. The sequence of frames is identical to that already described on page 9. Student errors in any frame do not jeopardize performance in the next frame. If a student does not figure the expected change correctly, the next frame draws the planned amount of change for the frame.

The practice test gives valuable practice, suggests the process encountered in real life, and reduces upcoming test anxiety. At the end of six store transactions, the student receives scores for abilities in adding coin values, knowing if he has enough money, figuring the lowest sufficient amount, subtracting and comparing change received with expected change.
The student's name, program title, number of items completed, and five skill scores are stored to disk.

The test program is identical to the practice test with the exception of data to be used in the store visit.

Although the student menu accepts only single letters to load a program, it will accept the secret code word 'menu.' When the computer automatically loads the student menu, teachers may type 'menu' to access student-protected programs. When 'menu' is typed, a teacher menu program is loaded. Options for the teacher include seeing all of the titles of the programs on the diskette, a brief description of each program, all money graphics shapes used in the program, load any student program or see student results.

By selecting to see all of the titles of the programs on the diskette the teacher can see how the entire product is organized, languages used, and size of the programs. The program which gives a brief description of every program on the diskette is loaded by the teacher menu. Designed as a convenience for teachers, it lists major purposes of each program and is useful should the written instructions be lost. Upon exiting this program, the teacher menu is loaded again.

By selecting to see the shapes used in the program, sixteen shapes and their numbers in the shape table are displayed. The program which draws the shapes loads a binary file called 'money shape table' which contains binary information used to draw any or all shapes. After the sixteen shapes are displayed another program is automatically loaded which uses a combination of shapes
to draw 1, 2, and 5 dollar bills. At the conclusion of the demonstration, the teacher menu is automatically reloaded.

By selecting to run a student program, the student menu program is loaded and the student menu displayed. As stated earlier, all programs reload the student menu upon their conclusion. After the teacher uses a student learning program, the menu will return. Upon typing the secret code 'menu' the teacher will return to the expanded teacher menu.

The final option of the teacher is the most useful. The 'teacher planning' program reads data from the diskette. Up to 50 student records can be stored which contain the student's name, program title, number of items completed, and the number of correct items. Any missed frames are printed with the student's incorrect answer and the correct answer. This applies to student learning programs one through six. Student records containing results of the placement exercises contain the recommended program number from which to start. Student records containing the results of the practice test or test will be expressed by printing the result in the skill areas.

The teacher may elect to see the results printed on the screen or produce a paper copy on a printer. The teacher may elect to save the current files for further use or delete them to make room for new ones.

Knowledge and Skills Gained

The programming and packaging experiences were particularly
useful. The use of menus, hello program, and loading programs under program control helped show how to increase user convenience and reduce user error. Knowledge of a fail-safe input routine will cause implementation of the routine in other products. The construction of shape tables can now be more rapidly and confidently done because of the amount of experience received in that area. Memory usage techniques will now be employed in other large programs as well as the use of hi-resolution graphics to print expanded text.

Gained also is the esoteric skill of sensing how long an educational planning or computer programming activity will take. An appreciation for the need of detailed planning has developed further, since small details overlooked caused backtracking and revision.

The intern's concept of use of the computer as an instructional aid has become more defined. Seeing neither total linear nor randomly generated programs as the answer to many situations, confidence in using the best of both approaches has increased.

A confidence that educational demands can be placed above programming demands has been developed. The resources of increased programming skills can be better applied to educational needs.

Observing the development of a product from start to completion has helped emphasize important times for developers to coordinate. The general outline of necessary programs, program features, types of stimulus and responses, and presentation of content are obvious points to coordinate. Experiences gained
emphasized the need to coordinate in a quantitative and qualitative manner. Proper coordination saves time and money, utilizes resources effectively, and ultimately insures a better product.
CHAPTER V

Conclusions and Recommendations

It becomes increasingly apparent that knowledge of the hardware and the DOS is vital to the creation of compact, powerful programs. Every effort must be made to understand the available machine language routines in residence and useful memory locations to best effect intended results.

Since many programs still break new ground in many ways, much time is spent in original work. More powerful subroutines and programmer's aids must be developed to minimize programming time. With the programming time reduced, more time will be allocated to planning and evaluating the instruction.

There is a definite need to develop communication links with curriculum planners and grass roots teachers. Program requests can never be assumed to reflect general or specific educational needs until a broader sampling of needs is conducted.

Arrangements have been made by the intern to receive field testing data and to become involved in the revision process. However, it would have been desirable to participate in the evaluation stages within the allotted time.

An internship involving technical skills and educational skills brings double rewards. Improving instruction and keeping pace with change are important goals for educators. These goals
can serve well any educational leader of the future.
APPENDIX A

Prospectus for Internship
Prospectus
for an
Internship
in
Computer Assisted Instruction

by

Jack H. Bender
Organization: Hartley Courseware, Inc.

Field Supervisors: Tim Hartley, programmer
                Jane Hartley, educational consultant

University Advisor: Dr. James R. Sanders, Western Michigan University

Major Focus of Experience: Design and implement a CAI module

Duration: Summer Term 1981 (240 hours)

Rationale

Microcomputers are encouraging a rebirth of individualized instruction and are facilitating record keeping, management and planning. With any new technology comes a need for many forms of technical and supervisory skills.

To adequately utilize a powerful tool such as the microcomputer, instruction designing skills and programming skills are needed. Many decisions will be better made through experiences which relate to feasibility, readability, usability, adaptability, and flexibility.

Working with a commercial firm has many benefits. The firm must be aware of current needs in software, industry standards, user amenities, level of user computer literacy, and testing procedures.

Description

Hartley Courseware specializes in microcomputer software for grades K-8. Most offerings are in math and language arts. Most
programs deal with single topics although any package offered will contain numerous programs of related topics. Appendix B provides a listing of program offerings.

It is anticipated that the internship can be used to observe and participate in the design and implementation of a software package. This experience will include initial educational planning, programming, and field testing.

Such an experience can help qualify an intern for placement in a supervisory position in a commercial firm or a school system.

Projected Nature of Internship Experience

<table>
<thead>
<tr>
<th>Objectives</th>
<th>Experiences and Contacts</th>
<th>Terminal Skills</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A. Conceptual</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. To review the availability of software on the project's topic</td>
<td>Examine catalogs and advertisements</td>
<td>Report the degree of need for software</td>
</tr>
<tr>
<td>2. To acquire a method for project definition</td>
<td>Discuss or observe procedures for defining the project</td>
<td>Apply a method for defining and describing the project</td>
</tr>
<tr>
<td>3. To become familiar with audience characteristics</td>
<td>Discuss scholastic abilities and degree of computer literacy the audience is assumed to possess</td>
<td>Design material appropriate to student skill level</td>
</tr>
<tr>
<td>4. To investigate an effective instructional planning model</td>
<td>Examine or observe models used in developing instruction</td>
<td>Design instruction using a planning model</td>
</tr>
<tr>
<td>Objectives</td>
<td>Experiences and Contacts</td>
<td>Terminal Skills</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>5. To acquire commercial standards for a complete software package</td>
<td>Discuss and observe software features which constitute a software package</td>
<td>Implement features which constitute a self-contained software package</td>
</tr>
<tr>
<td>B. Technical</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. To gain methods of handling input/output errors</td>
<td>Read computer program statements which effectively deal with errors</td>
<td>Implement error handling routines in all computer programs</td>
</tr>
<tr>
<td>2. To investigate effective utilization of memory</td>
<td>Discuss ways in which large programs can reside in limited memory</td>
<td>Demonstrate the use of memory management techniques</td>
</tr>
<tr>
<td>3. To understand the use of graphics to present expanded text</td>
<td>Read computer program statements which facilitate the display of text on the graphics screen</td>
<td>Develop routines which print text in the graphics mode</td>
</tr>
<tr>
<td>4. To obtain a means of copy protection for copyrighted material</td>
<td>Discuss program protection measures</td>
<td>To utilize program copy protection procedures</td>
</tr>
<tr>
<td>5. To acquire a means of determining reading level of textual presentation</td>
<td>Observe or examine methods for determining text readability levels</td>
<td>Predict and describe text readability levels for textual material</td>
</tr>
<tr>
<td>C. Human</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Understand the separate role of programmer and educational consultant</td>
<td>Observe activities of programmer and educational consultant</td>
<td>Describe key responsibilities of a programmer and educational consultant</td>
</tr>
<tr>
<td>2. Understand the necessary points for interaction between the programmer and educational consultant</td>
<td>Observe and discuss demands for joint planning</td>
<td>Recognize important decisions which must be coordinated</td>
</tr>
<tr>
<td>Objectives</td>
<td>Experiences and Contacts</td>
<td>Terminal Skills</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>3. Understand the software needs of schools as they relate to learner problems</td>
<td>Discuss how the needs for programs are uncovered</td>
<td>Describe sources and decisions for developing products</td>
</tr>
<tr>
<td>4. Understand strategies used with students during field testing</td>
<td>Observe procedures which determine effectiveness of the product</td>
<td>Describe the evaluation process used with individual students</td>
</tr>
</tbody>
</table>
APPENDIX B

Course Offerings of Hartley Courseware
All Hartley courseware requires a 48k Apple II with Applesoft in ROM and one Disk Drive.

### READING–LANGUAGE ARTS

<table>
<thead>
<tr>
<th>Title</th>
<th>Appropriate Grade Level</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vowels</td>
<td>1 - 3</td>
<td></td>
</tr>
<tr>
<td>Consonants/Blends</td>
<td>1 - 4</td>
<td></td>
</tr>
<tr>
<td>Letter Recognition</td>
<td>K - 1</td>
<td></td>
</tr>
<tr>
<td>Multiple Skills</td>
<td>1 - 3</td>
<td></td>
</tr>
<tr>
<td>WORDSEARCH</td>
<td>1 - 8</td>
<td></td>
</tr>
<tr>
<td>Roots/Affixes</td>
<td>3 - 8</td>
<td></td>
</tr>
<tr>
<td>Homonyms</td>
<td>3 - 8</td>
<td></td>
</tr>
<tr>
<td>Antonyms/Synonyms</td>
<td>3 - 8</td>
<td></td>
</tr>
<tr>
<td>Verbs</td>
<td>2 - 8</td>
<td></td>
</tr>
<tr>
<td>Nouns/Pronouns</td>
<td>3 - 8</td>
<td></td>
</tr>
<tr>
<td>Vocabulary–Dolch</td>
<td>K - 3</td>
<td></td>
</tr>
<tr>
<td>Vocabulary–Elementary</td>
<td>1 - 4</td>
<td></td>
</tr>
<tr>
<td>Spell It</td>
<td>1 - 8</td>
<td></td>
</tr>
<tr>
<td>Word Families</td>
<td>1 - 4</td>
<td></td>
</tr>
</tbody>
</table>

### MATHEMATICS

<table>
<thead>
<tr>
<th>Title</th>
<th>Appropriate Grade Level</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Prescriptive Math Drill</td>
<td>1 - 4</td>
<td></td>
</tr>
<tr>
<td>Number Words–Level 1</td>
<td>K - 2</td>
<td></td>
</tr>
<tr>
<td>Number Words–Level 2</td>
<td>2 - 5</td>
<td></td>
</tr>
<tr>
<td>Calendar Skills</td>
<td>2 - 5</td>
<td></td>
</tr>
<tr>
<td>Expanded Notation</td>
<td>2 - 5</td>
<td></td>
</tr>
<tr>
<td>Clock</td>
<td>1 - 5</td>
<td></td>
</tr>
<tr>
<td>Metric Drill</td>
<td>4 - 8</td>
<td></td>
</tr>
<tr>
<td>Integers/Equations</td>
<td>6 - 8</td>
<td></td>
</tr>
<tr>
<td>Math Concepts</td>
<td>2 - 8</td>
<td></td>
</tr>
</tbody>
</table>

### CREATE YOUR OWN COURSEWARE

<table>
<thead>
<tr>
<th>Title</th>
<th>Appropriate Grade Level</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE–Fill in the Blanks</td>
<td>K - 6</td>
<td></td>
</tr>
<tr>
<td>CREATE–Spell It</td>
<td>1 - 6</td>
<td></td>
</tr>
<tr>
<td>CREATE–Vocabulary</td>
<td>1 - 6</td>
<td></td>
</tr>
<tr>
<td>CREATE Skills–Elementary</td>
<td>1 - 6</td>
<td></td>
</tr>
<tr>
<td>CREATE Skills–Intermediate</td>
<td>4 - 8</td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX C

Available Software
Software available from
K-12 MicroMedia*
P.O. Box 17
Valley Cottage, N.Y. 10989

Change Maker (3-7) Provides practice in this real-world survival skill based on randomly selected purchase and payment amounts, but with immediate reinforcement for each answer.
8K PET; 16K II
MLW-PET-5779
MLW-TRS-5780

Learning to Count Money (4-6) Three programs teach the survival skill of counting money in a fun and unforgettable way. One program is a tutorial that requires a fourth grade reading level. In program two, the student goes on a Shopping Trip and will buy things at various stores, needing to collect 1000 purple stamps to win. In Check-Out the student runs a checkout counter. Superior graphics throughout. Includes three program tapes and an additional documentation tape.
16K II
MER-TRS-5691

Piggy Bank (4-8) A coin-counting game to improve students' skill in making change. Good sound reinforcement. Kids get to keep the computer coins if they're right.
16K
CAP-TRS-5026

Evaluations of Software available for Microcomputers

MACUL Journal*
Oakland Schools
2100 Pontiac Lake Rd.
Pontiac, MI 48054

Making Change (3 up). Making Change (also called Cash Register when run) explains how to make change by two methods. The first is by subtracting the amount owed from the amount received for the amount of change. The second is to start with the amount owed, for example 57 cents, and add 3 pennies to get 60 cents. You continue to add another nickel, dime, and quarter to add up to the total amount of 1 dollar received. Drill problems are then given to reinforce the second method.

This program is well done and would be useful in learning to make change. Some of the format of presentation could be modified to make the information more legible by skipping lines between printed lines of information. Another helpful addition would be to allow the student more control over the speed of the presentation by using the space bar to continue.

Ratings: (1 = poor to 5 = best, 0 = not available)

1. Program polish 4
2. Use of machine capability 3
3. Written documentation 0
4. Ease of use 4
5. Educational value 4

Computer: Apple-16K-INT
Availability: MACUL Math Vol. 1
Special requirements: Disk

Change (3 up). Change simulates what it would be like to be making change at a check-out counter. You are given the amount of the purchase and the amount that is given to pay for the purchase. The computer then uses high resolution to draw a cash drawer with locations for all coins and bills. Starting with a question mark in the penny section of the cash drawer, the program asks how many pennies are needed. The number you enter is transferred to the penny section and you move on to nickels and so on until up to the $20 bill is considered. If you are correct, the computer

prints a large "very good" positive reinforcement using the high
resolution graphics. This would be both fun and informative for
elementary and junior high students.

Ratings: (1 = poor to 5 = best, 0 = not available)

1. Program polish 4
2. Use of machine capability 4
3. Written documentation ?
4. Ease of use 4
5. Educational value 4

Computer: Apple-32K-AS
Availability: MECC Elem. Vol. 1
Special requirements: Disk
APPENDIX D

Definition of Terms
Definition of Terms

ASCII — "American Standard Code for Information Interchange." The numerical representation which computers use for numerals, letters, and characters.

BASIC — "Beginning All-Purpose Symbolic Instruction Code." Most common language used by people to program computers.

BINARY — Number system whereby each digit stands for a power of 2 as opposed to the usual decimal system which uses powers of 10.

BIT — The smallest amount of information that can be known. The choice between 2 alternatives, i.e., yes or no, true or false . . .

BYTE — 8 bits. Used to represent an alphanumeric character or number between 0-255.

COMMAND — A request to the computer that is executed as soon as it is received.

Computer — Any device that can receive and follow instructions to manipulate both numbers and letters and whose instructions may be changed.

CRT — "Cathode Ray Tube" — T.V. set.

Debug — To find errors and correct them.

Diskette (floppy) — Disk in square plastic envelope. 13 or 20 cm on a side.

Disk Drive — A peripheral which can store and retrieve information from a disk.

DOS — "Disk Operating System." Collection of disks which makes use of disk drive possible.

Graphics — Ability to use the computer to draw graphs, charts, figures, scenes, and simulated objects.

Hardware — The physical parts of a computer.

Hexadecimal numbers — Numbers which are expressed in base 16.

Hi-Resolution graphics — Graphics made which divide the screen into small parts, allowing for drawing of small detailed objects.
I/O — "Input and/or Output."

K - 48K — "Kilo" or one thousand (1024 to computer users). 
48K = 1024 x 48 bytes.

Listing — A list of program statements sent to the TV screen 
or printer.

Menu — List of options from which to choose.

Microcomputer — Computer based on a microprocessor.

Microprocessor — An integrated circuit that performs the task of 
executing instructions.

Personal computer — A moderately priced computer intended for 
general use by an individual.

Programmer — A person who writes programs.

RAM — "Random Access Memory." A computer memory where informa­ 
tion and programs are stored.

ROM — "Read-Only Memory." Memory which stores information once 
(i.e., by a manufacturer) and cannot be changed.

Shape table — A numerical set of instructions used in drawing 
shapes on the hi-resolution screen.

Software — Programs.

Statement — An instruction.

Strings — Sequence of letters, numerals, and characters.

Subroutine — Part of a program that can be performed by a 
special statement.

Text — Data other than numbers.

Variable — A quantity—place where a value may be stored.
APPENDIX E

Shape Example
To create a shape, the following procedures are used:

1. Draw the shape on graph paper
2. Select a starting position
3. Lay out vector arrows
4. Code the vectors into binary form
5. Code the binary form into hexadecimal form
6. Enter the data into memory
7. Save to diskette

There are eight vectors possible:

<table>
<thead>
<tr>
<th>No plot</th>
<th>Code</th>
<th>Plot</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>↑</td>
<td>000</td>
<td>↑</td>
<td>100</td>
</tr>
<tr>
<td>→</td>
<td>001</td>
<td>→</td>
<td>101</td>
</tr>
<tr>
<td>↓</td>
<td>010</td>
<td>↓</td>
<td>110</td>
</tr>
<tr>
<td>←</td>
<td>011</td>
<td>←</td>
<td>111</td>
</tr>
</tbody>
</table>

Here is a diagram showing the start position and some of the vectors:

```
    C
   ...
 start
   ...

 etc.
```

Here are two examples of binary and hexadecimal forms:

<table>
<thead>
<tr>
<th>binary form</th>
<th>hexadecimal form</th>
</tr>
</thead>
<tbody>
<tr>
<td>011100</td>
<td>$1C</td>
</tr>
<tr>
<td>110111</td>
<td>$37</td>
</tr>
</tbody>
</table>
\begin{tabular}{|l|l|}
\hline
\textbf{binary form} & \textbf{hexadecimal form} \\
\hline
0001 1110 & $\$1E$ \\
0011 0111 & $\$37$ \\
0001 1010 & $\$1A$ \\
0010 0100 & $\$24$ \\
0011 1111 & $\$3F$ \\
0001 1100 & $\$1C$ \\
0001 1100 & $\$1C$ \\
0010 0100 & $\$24$ \\
0000 1100 & $\$0C$ \\
0000 1100 & $\$0C$ \\
0010 1101 & $\$2D$ \\
0010 0100 & $\$24$ \\
0001 0101 & $\$15$ \\
0010 1010 & $\$2A$ \\
0001 0101 & $\$15$ \\
0001 1111 & $\$1F$ \\
0011 0011 & $\$33$ \\
0011 0110 & $\$36$ \\
0010 1110 & $\$2E$ \\
0000 0000 & $\$00$ \\
\hline
\end{tabular}
APPENDIX F

Sample Frames
For all money programs you will use one of three ways to answer.

One way is like this example. Just read the words and press the space bar.

press 'space bar' to go on

The 3rd way is to point to the answer then use 'return'. Use the arrow keys to point. When the hand is pointing to the correct answer - press 'return'. Try it now.

a) wrong

b) wrong

c) correct
Cost is 20 cents.

You have

Can you buy the item?

a) no

b) yes
On the back of the penny is the Lincoln Memorial.
Look under the memorial.

The value of the penny is ....

- a) one dime
- b) one cent
- c) one coin
- d) one quarter

Point to the coin.

- a) [Diagram of a coin]
- b) PENNY
You have $0.50.

If the item costs $0.20 the amount you should give is
a) $0.05
b) $0.10
c) $0.25
d) $0.50

This is the money you have.

How much do you have?

a) $0.25
b) $0.35
c) $0.50
d) $1.00

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
RUN

CHOICES ARE:

1 - RUN LESSONS
2 - STUDENT PLANNING
3 - SEE PROGRAM DESCRIPTIONS
4 - DEMO COIN SHAPES
5 - SEE CATALOG
6 - QUIT

ENTER SELECTION (1-6)
AND PRESS RETURN

STUDENT: JACK BENDER
MONEY: QUIZ

SCORE OF 1-3 MEANS PRACTICE NEEDED.
COUNTING MONEY - 1
KNOWING IF ONE HAS ENOUGH MONEY - 1
GIVING JUST ENOUGH TO BUY THE ITEM - 1
KNOWING WHAT CHANGE TO EXPECT - 1
(SUBTRACTION)
BIBLIOGRAPHY


Hartley Courseware Catalog. Dimondale, MI.


Milner, S. D., Mason, G., & Hargan, C. "Microcomputers ... the future is now," The Practitioner, October 1979.

Moursund, D. The Computing Teacher, 1980, 8(4), 1-76.

Nelson, T. Creative Computing, 1980, 6(9), 1-224.

Peters, H. J., & Johnson, J. W. Author's guide, design, development, style, packaging, review. CONDUIT. Iowa City, Iowa: The University of Iowa, 1978.


