Review
Reviewed Work(s): Introduction to Apple II BASIC by Ron M. Lavine
Review by: Jeff Bloom
Source: *The Arithmetic Teacher*, Vol. 33, No. 1 (September 1985), p. 53
Published by: National Council of Teachers of Mathematics
Stable URL: https://www.jstor.org/stable/41192679
Accessed: 25-03-2022 03:02 UTC

This package is targeted for first graders and younger children not ready for formal instruction and practice in reading; its six programs provide practice in discriminating shapes, letters, and numerals.

The format is similar in all six programs. Four programs ("Beam Up," "Alphasaurus," "ABC," and "Moon Math") merely require children to push any key when two identical shapes, letters, or numerals appear simultaneously on the screen. Correct responses receive visual and auditory reinforcement (e.g., an entertaining chorus line of baby chicks high-stepping to a catchy tune); incorrect ones receive a beep. The other two programs extend beyond this perceptual matching procedure. "Letter Getter" pairs uppercase and lowercase letters, whereas "Number Chick" pairs numerals with a die showing the correct side.

From a technical standpoint, the programs do have several strengths. The screen is uncluttered; and the objects, from baby chicks to spaceships, are recognizable and presented in bright, attractive, and unusual colors. The movement of the objects is smooth, and sound is used sparingly. The menu offers choices of one of the six games, directions, and end-session and change-game options. Directions (specific to each game) and end-session options provide useful guidance that the teacher can pass on to nonreading students. The change option gives the teacher or student opportunities to change the speed of presentation, the number of correct matches needed to complete a game, the letters to be used, and the order of presentation of numerals (sequential vs. random). As the authors point out, frequent use of the change option will enhance the effectiveness of the package.

These strengths aside, this material suffers from two major problems related to content. First, in an apparent attempt to make the programs gamelike, dubious themes are introduced that are conceptually confusing. Consider "Alphasaurus." A green sea serpent pushes up a letter, and a dinosaur emits one from its mouth. When the letters match, pushing a key causes one letter to shoot across and blow up the other letter. Why? In several instances, the aggressive and fantastic content typical of video games is imposed somewhat artificially and inappropriately on the goal of recognition of letters and numerals. Second, the goals and requirements of *Getting Ready to Read and Add* are very limited. Many kindergarten children, and even preschoolers, may find the simple matching required in most programs to be boring. And many teachers will recognize that, from most current viewpoints, letter and numeral recognition are but a small part of beginning reading and number use. Ultimately, *Getting Ready to Read and Add* is more than adequate technically but suffers from a narrowness of focus, thematic confusion, and unchallenging requirements for even young children.—*David E. Fernie, University of Houston, Houston, TX 77004.*

Introduction to Apple II BASIC. *Ron M. Lavine. 1984, student textbook, $7.11; teacher's resource material (including diskette), $48. Houghton Mifflin, One Beacon Street, Boston, MA 02108.*

This set is aimed at students in grades 6–8, although the publishers suggest that even high school students will not "find it beneath their level." The intent is to present a step-by-step, hands-on approach to BASIC programming.

The student text consists of twenty-six chapters that each represent a single lesson of twenty-five to forty-five minutes. The text is easy to read and understand. In addition, the illustrations and sample program segments are appropriate and helpful. Each chapter consists of an introduction to a specific set of statements followed by exercises on the computer. The same format is repeated with extended uses and ramifications of these statements. The chapters conclude with a lesson summary, review questions, and extras for experts. The first few chapters use the tutorial disk. The content covers, for the most part, what would be expected in an introductory course in BASIC: PRINT variations, INPUT, loops, formatting functions, graphics, algorithms, and so forth.

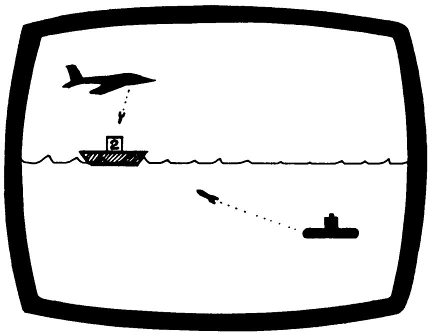
The most critical problem in the text is in the logical ordering of chapters or content. For instance, chapter 9 discusses "debugging programs." Two types of errors are discussed: syntactical and logical. The TRACE function is introduced as a useful debugging tool. The assumption is that students will understand the logical structure of programming at this point in the book. Chapter 23 (almost at the end of the book) discusses flowcharting and algorithms. For a beginning programmer, flowcharting is an invaluable activity for understanding the logical sequencing of programs. Without this previous knowledge, a description of the TRACE function seems to be somewhat useless. The chapter on flowcharting would be better placed near the beginning of the book. Other oddities in the sequence are these: (1) FOR . . . NEXT loops at the middle of the book, nine chapters before IF . . . THEN statements; (2) READ . . . DATA statements in chapter 19; and (3) lo-res graphics in chapter 11, before any of the functions just described. Another point worth mention is the introduction of high-resolution graphics so early in the book. Knowing the excitement this topic could create among the students is one thing, but what happens when they get so carried away that they use up all the memory space and their programs begin to disappear before their very eyes? Try to debug that one! Just think of the amount of space even a relatively short high-resolution program takes up, especially when logical programming has not yet been introduced!

The tutorial disk is not worth the space it takes up. The printing speed is so slow that even a slow reader is bound to get bored. User prompting would be much more reasonable and would probably cut the lesson time by a third. The same content is practiced (repeated), and the user is given no chance to skip ahead to the next content area. If the disk is designed to ensure that each student gets the point, it misses the mark. When the student is asked to correct a sentence or statement, the program does not check for correct responses. However, when it asks for the correct answer to a question, the student has to give the exact answer or there is no escape (except with the reset button and on-off switch). Even after repeated attempts, the correct response is never given.

In the teacher's guide, each chapter is headed by one or two objectives and followed by teaching ideas, pointers for programming, an idea bank, and answers to review questions. The ideas are fairly obvious to a somewhat experienced teacher. The programming pointers mention some of the more obvious mistakes that can occur and some further activities to reinforce the ideas being presented. The idea bank presents more of the same. At the end of the teacher's guide are some useful blackline drawings and grids that can be reproduced for use by students. To the teacher with a reasonable background in BASIC, this guide is not of much use.

The student text is worth a look. I would definitely recommend rearranging the sequence of chapters and possibly leaving out the chapter on high-resolution graphics. As a supplementary or enrichment text, it could be worth considering. Students could work through it with very little supervision and probably enjoy themselves in the process. As a primary text, the teacher should be prepared to do some further preparation, in addition to rearranging the sequence. Teachers may want to plan class presentations rather than have the students read in class. And since only seven error messages are mentioned in the text, it would not be a bad idea to think about the potential problems beforehand.—*Jeff Bloom.*

Math Wars. *Marley W. Watkins. 48K, Apple II$^+$, IIc, or IIe with disk drive. 1982, $39.95. One diskette and teacher's guide. Free backup disk. SouthWest EdPsych Services, P.O. Box 1870, Phoenix, AZ 85001.*



This educational program with color graphics, for one or two players or teams, was developed to engage students in requisite mathematics practice while they play a challenging and entertaining arcade game. Teams or individuals can choose to be either the navy or the air force. The goal is to determine which service can successfully destroy an enemy ship that is carrying the correct responses to randomly generated arithmetic problems. The content covers grade levels K–9 in addition, subtraction, multiplication, and division of whole numbers, fractions, and decimals. As with many