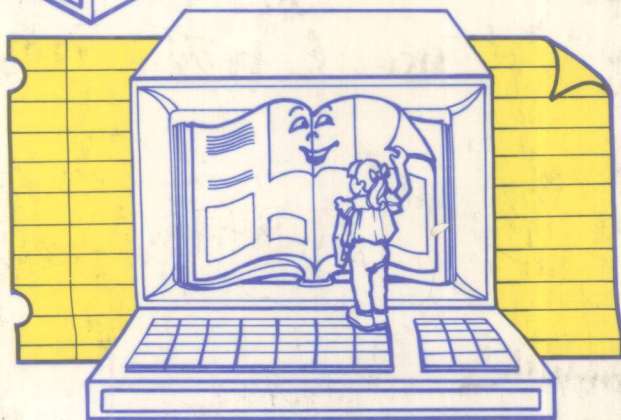
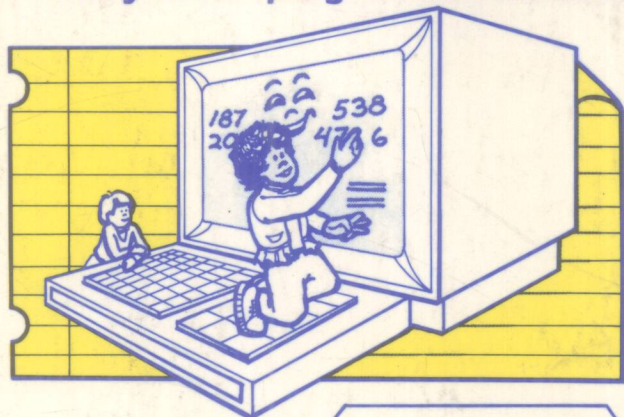


TRS-80

Teaching Aid

ready-to-run programs for the classroom and home



Edward Burns

G434
B4

8566011



TRS-80 TEACHING AID

Ready to Run Programs
for the Classroom
and Home

Edward Burns



a *Reston Computer Group* book
Reston Publishing Company, Inc.
a *Prentice-Hall* company
Reston, Virginia

Library of Congress Cataloging in Publication Data

Burns, Edward,
TRS-80 teaching aid.

“A Reston Computer Group book.”

Bibliography: p.

Includes index.

1. Computer-assisted instruction. 2. TRS-80 (Computer)
—Programming. I. Title. II. Title: T.R.S.-eighty
teaching aid. III. Series.

LB1028.5.B874 1984 371.3'3 83-24722

ISBN 0-8359-7877-X

ISBN 0-8359-7875-3 (pbk.)

© 1984 by **Reston Publishing Company, Inc.**
A Prentice Hall Company
Reston, Virginia 22090

All rights reserved. No part of this book may be reproduced, in any way, or by any means, without permission in writing from the publisher.

TRS-80[®] is a registered trademark of Radio Shack[®], a division of the Tandy Corporation.

10 9 8 7 6 5 4 3 2 1

Interior design and production: Jack Zibulsky

Printed in the United States of America

TRS-80 TEACHING AID

Ready to Run Programs
for the Classroom
and Home



E8566011

To Virginia Goss Burns

PREFACE

The purpose of this book is threefold:

- 1) To provide a series of computer programs relating to achievement, learning and educational processes.
- 2) To provide programs that could be entered and run within a relatively short period of time.
- 3) To provide programs that might suggest ways in which microcomputer systems could be used to meet specific educational needs.

These goals are not always compatible. For example, limiting a program to forty or so statements can detract from the program's overall sophistication. Still, there is a strong need for a series of programs that illustrate the vast potential of microcomputers in education—programs that do not require many hours of programming and debugging.

xii | Preface

I hope the programs in this book will help you effectively use microcomputers as teaching aids.

To my primary sources of ideas, I express my gratitude; Jon and Ted Burns have taught me more about achievement and learning than a book ever could.

E.B.

CONTENTS

PREFACE xi

1 INTRODUCTION, 1

- TRS-80 Models, 2
- Programming, 3
- Saving Programs, 5
- Bugs, 6
- The Goal, 7
- The Friendly Computer, 7

2 FEEDBACK, 9

- Descriptors, 9
- Personalized Feedback, 11

viii | Contents

Negative Feedback, 12
Graphics, 13
Conclusion, 15

3 READING, 17

—PRINT @ Statements, 18
Tachitoscope, 19
—Random Numbers, 22
Self-adjusting Tachitoscope, 22
—Motor and AUDIO Statements, 25
Spelling Practice, 26
Sequential Memory, 28
—ASCII Codes, 31
Letter Synthesis, 32
Syllable Synthesis, 34
Speed Reading, 38
—Color, 42
Left-right Letter Tracking, 43
—REM Statements, 45
Word Closure, 46
—Left-right Eye Pursuits, 51
Controlled Sentence Reading, 52
Language Builder, 54
—END and DATA Statements, 60
Foreign-language Tutor, 60
Touch Typing, 65
—Sentence Writer, 68

4 LEARNING, 69

Eye-hand Reaction, 70
Serial Letter Learning, 71
Digit Span, 74
Simon Says, 76
Time Perception, 79
—Key Speed, 81
Anagram Problem Solving, 82
Letter Discrimination, 84
Visual Search, 87
Number/Letter Memory, 89

- Coding, 92
 - The LEN Game, 95
- Free-recall Learning, 96
- Paired-associate Learning, 103
 - The SOUND Statement, 109
- Sound Discrimination, 109
 - The Micro Color Computer, 112
- Note Learning, 113
- Learning the Morse Code, 116
 - A PRINT Shortcut, 119

5 MATHEMATICS, 121

- Primary Facts, 122
- Remedial Math Practice, 128
- Specific Fact Practice, 131
 - Compacting, 135
- Graphic Facts, 135
 - Decimal Places, 140
- Repeated Addition, 140
- Division Practice, 143
- Math Word Problems, 146
- Learning to Regroup, 148
- Number Sequence, 152
 - The GOSUB Statement, 154
- Basic Fraction Skills, 155
- Reducing Fractions, 159
- Adding Fractions, 161
 - Number Guess, 167

6 MISCELLANEOUS, 169

- Test Statistics, 170
- Stanines, 174
- Percentile Ranks, 176
 - Scientific Notation (E Notation), 178
- Random Ranks, 179
- Rank-order Correlation, 183
- Simulation, 185
- Predicting Grade Equivalents, 188
- Predicting School Enrollments, 189

x | Contents

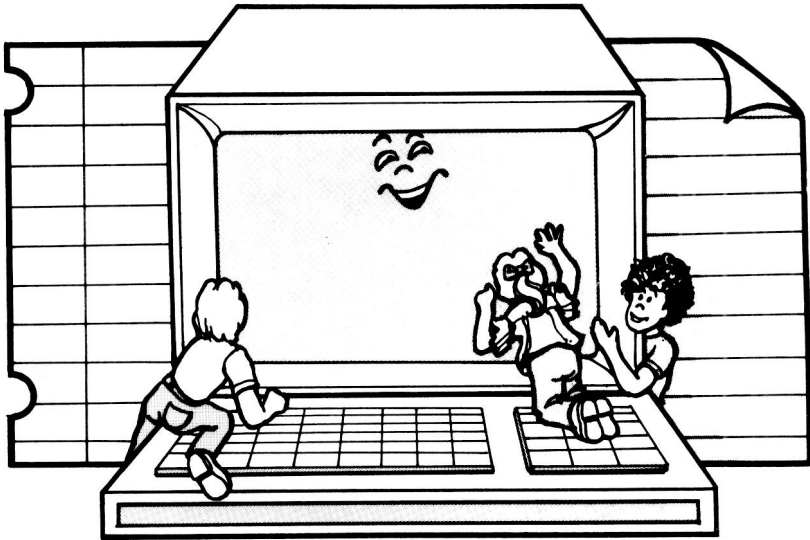
- A Normal Curve Experiment, 191
- Sort, 193
 - Memory, 195
- Readability, 196
- The Flesch Formula, 199
- Learning Programming Logic, 205
- Special Math Numbers, 209
 - Squares and cubes, 209
 - Prime numbers, 209
 - Square roots, 210
 - Factorials, 211
 - Fibonacci numbers, 212
 - Pythagorean triples, 212
 - Binary numbers, 213
- The Lotto Game, 214
- Dice Game, 215

BIBLIOGRAPHY, 219

INDEX, 221

1

— INTRODUCTION —



Microcomputers can be used to perform a variety of educational tasks. The purpose of this book is to illustrate how TRS-80® microcomputer systems can be used to teach reading and math skills, to improve learning, and to solve a variety of educational problems. Emphasis is placed on the word *illustrate* because no program, or series of programs, is suitable for all needs. The beauty of computer programming is that programs can be designed to meet specific, individual needs. If a program in this book does not meet a specific need, by all means change the program. When using a microcomputer system, especially with children, the goal is straightforward: use programs that solve specific problems; don't use programs simply because they are available.

The programs in this book relate to the development of specific reading and arithmetic skills, psychological learning and memory processes, and the solving of educational problems. There is no order for

2 | INTRODUCTION

using the programs. First determine exactly what you want the computer to do, and then find a program that will best accomplish that task.

The programs in this book are arranged in five chapters: *Feedback*, *Reading*, *Learning*, *Mathematics*, and *Miscellaneous*. The reading programs include a variety of tachistoscopes, spelling, language, and reading comprehension programs. The learning programs illustrate a variety of psychological tasks with special emphasis on memory. The mathematics section offers methods and techniques for teaching primary facts, computational skills, and fractions. Included in the miscellaneous are programs to calculate statistics, to determine test norms, and to perform a variety of educational tasks.

Each program described is accompanied by a general description, a program description, a complete program listing, and one or more samples showing how the program is used and the type of output to expect. An attempt has been made to outline the important characteristics of each program in terms of use and programming format. When appropriate, possible program modifications are suggested.

All the programs have been written so that they are not excessively long and so that the logic of each is easily understood. The majority of programs have 35 to 45 statements with no compacting; that is, statements are not compacted on one line by means of a colon.

Periodically throughout the book you will encounter boxed notes. Each of these notes describes a specific programming concept or TRS-80 programming technique. The purpose of these brief notes is to illustrate interesting programming options or occurrences that are not covered more extensively in the text.

TRS-80 models

All of the programs presented in this book have been written in BASIC. Each program has been run using the simplest of all TRS-80 models: the 4K TRS-80 color computer. The programs can be used with both Extended and non-Extended Color BASIC. With several minor modifications, the programs can be used with noncolor TRS-80 models (e.g., the TRS-80 Model III).

The primary difference between color and noncolor TRS-80 models concerns the use of the SOUND and PRINT @ statements and the CHR\$ function. Noncolor computer models do not have a SOUND statement. However, because only a few of the programs in this book use the SOUND statement, this should be a minor limitation.

The `PRINT @` statement is used to begin printing at a specified screen location. The following statement instructs the computer to begin printing the word `HELLO` at screen position 42:

```
10 PRINT @ 42, "HELLO"
```

For the color computer models, screen location 42 is the 11th column of the 2nd line because the color computer uses a 32-column screen. For noncolor computer models, screen location 42 is the 43rd column of the 1st line because these models use a 64-column screen. For the most part the corresponding noncolor computer model screen location can be determined simply by multiplying the color computer screen location value by 2. If the color screen location is 42, then the corresponding noncolor screen location is 84.

In the first program provided in the *Reading* chapter, Tachistoscope, the words are flashed beginning at screen location 234:

```
80 PRINT @ 238,W$(R)
```

To flash words in the corresponding noncolor TRS-80 screen location, the preceding statement is changed to the following:

```
80 PRINT @ 476,W$(R)
```

With the `CHR$` function, the only difference entails the use of the color computer graphic character `CHR$(128)`, which is a solid square. The corresponding noncolor computer graphic character is `CHR$(143)`.

Programming

For those who are unfamiliar with programming or the TRS-80, the best way to get started is to jump in and try one of the programs.

After entering each line (or statement), press the `ENTER` key.

```
10 FOR X=1 TO 10
15 PRINT X
20 NEXT X
25 END
```

Once the program has been entered, type the word `RUN` and then press the `ENTER` key to run the program. If there are no bugs, the numbers 1 through 10 should be printed.

```
1
2
3
4
```

4 | INTRODUCTION

5
6
7
8
9
10

But what if there is a bug; that is, what if there is a mistake that causes the program not to run? If a program does not run, try to determine approximately where the mistake is in the program.

In the preceding program, assume that the following syntax error was made in statement 20:

```
10 FOR X=1 TO 10
15 PRINT X
20 MEXT X
25 END
```

When you run this program, the following will occur:

```
RUN
1
?SN ERROR IN 20
```

This means that a syntax error was encountered in statement 20. This error is corrected by simply retyping statement 20 and then running the program:

```
20 NEXT X
RUN
1
2
3
4
5
6
7
8
9
10
```

To review a program when checking for bugs, the LIST command is extremely useful. Type LIST and then press the ENTER key; the program will be listed on the screen.

```
LIST
10 FOR X=1 TO 10
15 PRINT X
20 NEXT X
25 END
```

You can also print everything up to a specific statement by specifying the statement number in the LIST command. In the following example, all the statements up to and including statement 120 will be printed:

```
LIST-120
```

Most bugs involve syntax errors; be sure that the statements are entered as shown (if you are using a noncolor TRS-80 computer, make the necessary adjustments for color and noncolor TRS-80 computer differences). Remember that the addition or deletion of a single character, even a single comma, can prevent an entire program from running, or cause it to run incorrectly.

Saving programs

Saving programs on tape by using a cassette recorder is a simple matter with the TRS-80. First, attach the recorder to the computer using the required cable (check the TRS-80 manual).

Rewind the cassette tape to the beginning. After the program has been debugged and is working as desired, the CSAVE command is used to save the program on tape. The following program, which is given the name DRAW, illustrates how the CSAVE command is used.

```
10 FOR X=1 TO 6
15 PRINT TAB(5)"H"TAB(14)"H"
20 NEXT X
25 PRINT TAB(5)"HHHHHHHHHH"
30 FOR X=1 TO 6
35 PRINT TAB(5)"H"TAB(14)"H"
40 NEXT X
45 END
```

The steps to be followed in saving the preceding program follow:

1. Run the program to be sure that it works as desired.
2. If the tape has no other programs on it, rewind the tape to the beginning. If the tape already has programs on it, skip to the end of the last program recorded. This is accomplished by using the SKIPF "NAME" command, where the name in quotation marks is the name of the last program entered.
3. Turn the volume about halfway (between 4 and 5) and press the record button.
4. Type the following to save the program:

```
CSAVE "DRAW"
```


6 | INTRODUCTION

This command causes the cassette recorder to record the program until it has been completely recorded.

The steps to be followed when loading a program from the cassette follow:

1. Use the SKIPF "NAME" command to find the end of the program immediately before the program that you wish to load.
2. Turn the volume between 4 and 5 and press the play button.

Once the program with the designated name is encountered, it will be loaded and ready to run.

Bugs

In many pursuits, making mistakes is frowned on. In the computer programming business, making mistakes is an occupational hazard. A person who has not had a bug or two in a program has probably not written many programs.

Two major categories of errors should be noted. The first category involves technical errors such as attempting to divide by 0 (an error), incomplete statements (such as a GOTO statement with no statement number after the GOTO), and problems of syntax. The TRS-80 manual lists all error messages and abbreviations. The more typical technical errors follow:

Abbreviation	Meaning
/0	Attempting to divide by 0
BS	Subscript out of range
I/O	Input/output error
NF	NEXT without FOR
OD	Out of data
SN	Syntax error
TM	Type mismatch
UL	Undefined line

The second category of bugs is errors involving logic. The computer does what it is instructed to do—exactly. Let us say that we want the computer to multiply a number by 10. We write a program that will read a number using an INPUT statement and then do the appropriate multiplication. We run the program, enter the number 5, and the computer prints 500. How can this be? Is the computer broken? Here is a listing of the program: