

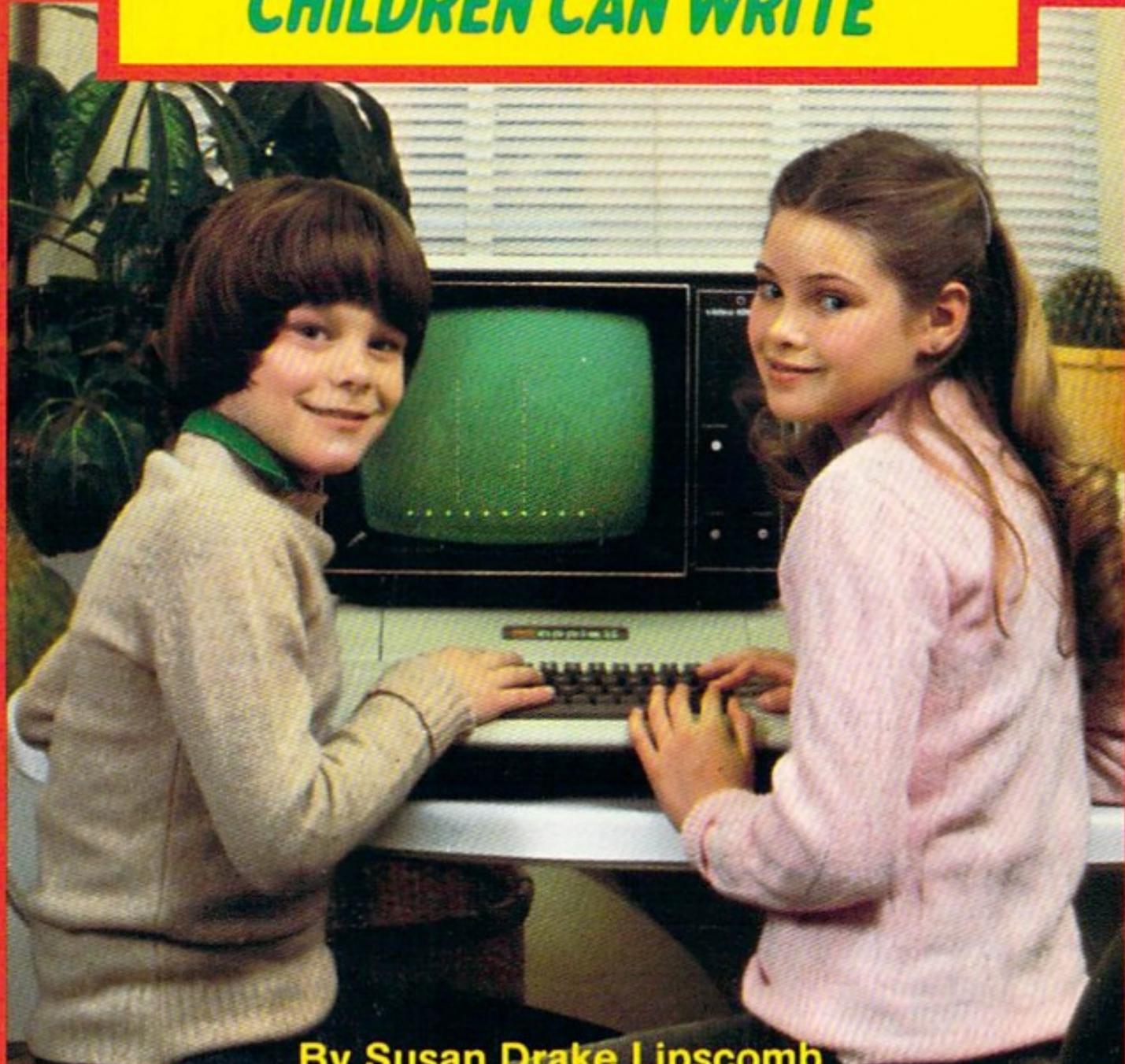
AVON **C**AMELOT

AVON-0-380-80606-1 \$2.95  
CAN. \$3.95

A Kids' Guide to  
BASIC Programming

# **BASIC FUN**

**COMPUTER GAMES,  
PUZZLES, AND PROBLEMS  
CHILDREN CAN WRITE**



By Susan Drake Lipscomb  
and Margaret Ann Zuanich

---

**BASIC FUN**

---

*Other Avon Camelot Books by*  
**Susan Drake Lipscomb and**  
**Margaret Ann Zuanich**

**BASIC BEGINNINGS**

**BASIC FUN WITH ADVENTURE GAMES**

**BASIC FUN WITH GRAPHICS: The Apple® Computer Way**

**BASIC FUN WITH GRAPHICS: The Atari® Computer Way**

**BASIC FUN WITH GRAPHICS: The IBM/PC® Computer Way**

**SUZAN DRAKE LIPSCOMB** and **MARGARET ANN ZUANICH** have a unique combination of skills that contributed to the creation of **BASIC FUN**. Margaret Zuanich's experience in the computer field has included everything from programming to management consulting. She earned her Master's Degree in Business and is now involved in computer systems training. Susan Lipscomb holds a Master's Degree in Education and has spent fourteen years in the area of language and learning disabilities. They both live in Palo Alto, California.

**Avon Books are available at special quantity discounts for bulk purchases for sales promotions, premiums, fund raising or educational use. Special books, or book excerpts, can also be created to fit specific needs.**

**For details write or telephone the office of the Director of Special Markets, Avon Books, Dept. FP, 1790 Broadway, New York, New York 10019, 212-399-1357. IN CANADA: Director of Special Sales, Avon Books of Canada, Suite 210, 2061 McCowan Rd., Scarborough, Ontario M1S 3Y6, 416-293-9404.**

---

**BASIC FUN**  
**COMPUTER GAMES,**  
**PUZZLES, AND PROBLEMS**  
**CHILDREN CAN WRITE**

---

**Susan Drake Lipscomb  
and Margaret Ann Zuanich**

AN AVON  CAMELOT BOOK

**BASIC FUN: COMPUTER GAMES, PUZZLES, AND PROBLEMS CHILDREN CAN WRITE** is an original publication of Avon Books. This work has never before appeared in book form.

6th grade reading level has been determined by using the Fry Readability Scale.

Front cover photograph shows Apple® II computer. Apple is a registered trademark of Apple Computer, Inc.

Book design by Joyce Kubat.

## AVON BOOKS

A division of  
The Hearst Corporation  
1790 Broadway  
New York, New York 10019

Copyright © 1982 by Susan Drake Lipscomb and Margaret Ann Zuanich  
Published by arrangement with the authors  
Library of Congress Catalog Card Number: 82-4057  
ISBN: 0-380-80606-1

All rights reserved, which includes the right to reproduce this book or portions thereof in any form whatsoever except as provided by the U. S. Copyright Law. For information address Avon Books.

First Camelot Printing, July, 1982

Library of Congress Cataloging in Publication Data

Lipscomb, Susan Drake.  
BASIC fun.

(An Avon Camelot book)

Summary: Introduces the fundamentals of computer programming using the BASIC programming language. Includes descriptions and listings of programs designed to amuse and challenge users.

1. Games—Computer programs—Juvenile literature.
2. Games—Data processing—Juvenile literature.
3. Basic (Computer program language)—Juvenile literature. [1. Electronic digital computers—Programming. 2. Basic (Computer program language). 3. Games—Computer programs. 4. Computers] I. Zuanich, Margaret Ann. II. Title. III. Title: B.A.S.I.C. fun.

GV1469.2.L57 794 82-4057  
ISBN 0-380-80606-1 AACR2

CAMELOT TRADEMARK REG. U. S. PAT. OFF. AND IN  
OTHER COUNTRIES, MARCA REGISTRADA, HECHO EN U. S. A.

Printed in the U. S. A.

OP 16 15 14 13 12

**We are indebted to the Palo Alto  
Unified School District Computer Services  
and the students and staff at  
Loma Vista Elementary School.**



---

# TABLE OF CONTENTS

---

Introduction	11
<b>1 PRINT: How the Computer Speaks</b>	<b>13</b>
Problems	
1. Poem	14
2. Flag	15
3. Love	16
4. Face	17
5. Stellar	18
6. Space	19
7. KNOCK KNOCK	21
8. Riddles	22
<b>2 LET: Adding 2 and 2</b>	<b>23</b>
Problems	
1. Recycle	24
2. Baby Sister	26
3. Paper Route	27
4. Retirement	28
5. Moon	29
6. Pizza	30
7. Swim Team	31
8. Inheritance	32
9. Sleep-over	33
<b>3 GOTO: Showing the Way</b>	<b>35</b>
Problems	
1. Stars	36
2. Counting	38
3. Power	39
4. Hunt	40
5. Gambler	42
6. Family Tree	44

## **4 INPUT: Talking to the Computer 46**

### Problems

1. Conversion 47
2. Silicon Valley Game 48
3. The Great Magician 50
4. Batting Averages 52
5. Year 2000 54
6. Averages 55
7. Savings Account 56
8. Mileage 58
9. Recipe 60

## **5 STRINGS: Using Words with the Computer 62**

### Problems

1. Nonsense 63
2. A Man from Peru 64
3. A Young Lady Named Maude 65
4. Race 66
5. Rhyme 68
6. Names 70
7. Author 71

## **6 IF . . . THEN: How the Computer Makes Decisions 73**

### Problems

1. Tax 74
2. Foreign Currency 76
3. Mystery Code 78
4. Ten Speed 79
5. Time Differences 81
6. Interior Decorating 83
7. Diver's Test 85
8. Spelling 87
9. You Are a Detective 89

**7 FUNCTIONS (INT, RND, ABS): Doing Complex Things the Easy Way 92**

Problems

1. Soccer 93
2. Shell Game 95
3. Dice Game 97
4. Number Guessing Game 100
5. Number Sequences 103
6. How Warm Is Your Heart? 105
7. War 107

**8 FOR . . . NEXT: The Computer Repeats Itself 110**

Problems

1. Fancy Counting 111
2. Blast Off 112
3. Pinball 113
4. Paper, Rock, Scissors 115
5. Ring the Bell 118

**9 FANCY PRINTING (TAB): Writing It Your Way 121**

Problems

1. Rectangle 122
2. Design-a-Scarf 124
3. HIHO 126
4. Lifeguard 127
5. Shape 129
6. Party 131
7. Fleet 133
8. Award 135

**10 SUBSCRIPTS: Where to Store Your Information 137**

Problems

1. Most Valuable Player 138
2. Bank Statement 141
3. Pascal's Triangle 143
4. Sea Battle 145

**11 SUBROUTINES: How the Pros Do It 149**

Problems

1. Birthday 150
2. Math Quiz 153
3. Tennis 157

**12 GAMES: Now You Can Do Your Own Thing 163**

Problems

1. Ski Weekend 164
2. Backpack 170

Appendix A 175

---

# INTRODUCTION

---

Are you getting bored with running packaged computer games? You can design your own once you learn how to communicate with the computer. *BASIC Fun* shows step-by-step how you can write computer programs—to solve your own problems, print your own pictures and invent your own games.

Special languages are used to communicate with a computer. For micro-computers, the language is called micro-BASIC. Each word in this language is called a programming statement. Computer programs are made up of programming statements which tell the computer what you want it to do.

*BASIC Fun* teaches you micro-BASIC by giving you simple programs showing how different programming statements work and where they belong in a program. First you should type and run the programs as they are printed in the book. Then see if you can make the few changes we've suggested, or use your own ideas to check your understanding of a programming statement. Finally, try to design your own programs or games just using ours as a guide.

**Warning!** You should use all the programs within each chapter—or at least read and understand those you don't use—before moving on to the next chapter. Following chapters build on and expand ideas learned in earlier ones—so work through the programs in the order they appear in the book.

Above all, we hope you have as much fun exploring your computer's possibilities and inventing programs as we did in writing this book.

The programs included in this text can be used on any computer that uses the programming language BASIC. They

were written in Applesoft BASIC using the Apple II Plus and tested on the Atari, Radio Shack TRS-80, Texas Instruments TI-99/4A, Hewlett-Packard HP-2000, IBM Personal Computer and the Osborn. We used a standard version of BASIC which is compatible with all of these computers. However, there are a few areas in which the computers differ. We have indicated in the Computer Notes following relevant programs any changes necessary to use the program on a particular computer. The Appendix contains a more detailed discussion of these differences. Before you begin using the programs, read the Appendix, and if necessary, refer to the Programmer's reference manual for your computer.

In order to make the programs as general as possible, we avoided using certain features included in one version of BASIC but left out in others. Once you become familiar with your own computer, you can add enhancements provided by your particular version of BASIC.

If you have trouble making a program work on your computer, we recommend the following steps:

1. Check your program line-by-line against the listing in the book.
2. Make sure you have made the changes for your computer given in the Computer Notes following each program.
3. Check the BASIC manual for your computer for any changes required for your version of BASIC.

If you still cannot make a program work on your computer, and think it is a genuine bug, send a description of the problem and a self-addressed, stamped envelope to the publisher. We will respond as quickly as we can to all such correspondence.

---

# 1 PRINT

## How the Computer Speaks

---

This programming statement allows you to make the computer type out anything you want on the screen or paper.

### PROBLEMS

1. Poem
2. Flag
3. Love
4. Face
5. Stellar
6. Space
7. KNOCK KNOCK
8. Riddles

---

# POEM

---

You can turn your favorite poem into a computer program. Then, whenever you want to read your poem, just run your program.

Here is a poem we wrote. Put this in your computer by typing each line exactly as shown in the Program Listing.

Once you have our poem working on your computer, try writing one of your own.

## SAMPLE RUN

```
ROSES ARE RED  
VIOLETS ARE BLUE  
YOU CAN LEARN TO WRITE PROGRAMS  
FOR MOST THINGS YOU DO
```

## PROGRAM LISTING

```
100 PRINT "ROSES ARE RED"  
110 PRINT "VIOLETS ARE BLUE"  
120 PRINT "YOU CAN LEARN TO WRITE PROGRAMS"  
130 PRINT "FOR MOST THINGS YOU DO"  
140 END
```

Note: Lines 100-130: When this program is run, the computer prints out every letter between the two " marks on each line.

Line 140: The end statement tells the computer that it is the last line in the program. This statement is required for some computers and optional for others. We have included it in this text so the programs can be used on most computers.



---

# LOVE

---

This program prints a very familiar word.

You can do the same thing with your own name or initials.

## SAMPLE RUN

```
X      X      X  X  XXXX
X     X X     X  X  X
X    X  X    X  X  X
X   X  X    X  X  X
X  X  X    X  X  XXXX
X X  X    X  X  X
X  X  X    X  X  X
XXXX  X      X  XXXX
```

## PROGRAM LISTING

```
110 PRINT " X      X      X  X  XXXX"
120 PRINT " X     X X     X  X  X"
130 PRINT " X    X  X    X  X  X"
140 PRINT " X   X  X    X  X  X"
150 PRINT " X  X  X    X  X  XXXX"
160 PRINT " X X  X    X  X  X"
170 PRINT " X  X  X    X  X  X"
180 PRINT " X  X  X    X  X  X"
190 PRINT " XXXX  X      X  XXXX"
200 END
```

---

# FACE

---

This program draws a person's face.

See if you can change the expression, or add some hair.

What happens if you forget a " when you are typing a PRINT statement?

## SAMPLE RUN



## PROGRAM LISTING

```
130 PRINT "  
140 PRINT "  
150 PRINT "  
160 PRINT "  
170 PRINT "  
180 PRINT "  
190 END
```



---

# STELLAR

---

This program draws a very familiar constellation of stars.  
Try writing a program that draws a different constellation.

## SAMPLE RUN

```
      *
    *
 *      *      *      *
      *

```

THIS IS A DRAWING OF THE BIG DIPPER

## PROGRAM LISTING

```
100 PRINT "
110 PRINT "
120 PRINT "      *"
130 PRINT "      *"
140 PRINT "*"
150 PRINT "
151 PRINT "
152 PRINT "
160 PRINT "
165 PRINT "
166 PRINT "
170 PRINT "THIS IS A DRAWING OF THE BIG DIPPER"
180 END
```

Note: Lines 165 and 166: The PRINT statement without any " " prints a blank line.

---

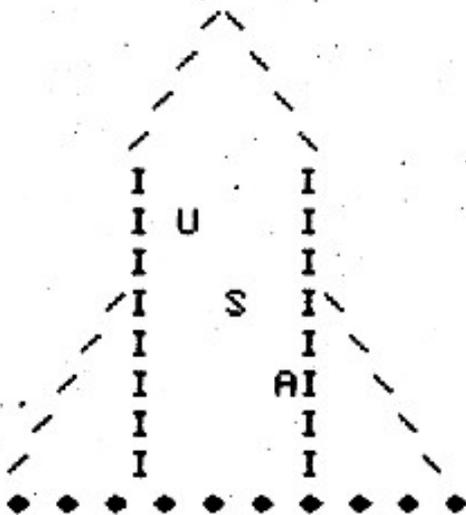
# SPACE

---

This program draws a U.S. rocket ship which blasts off from your video terminal.

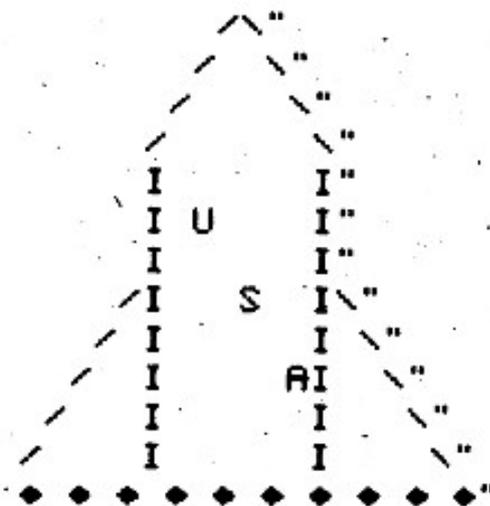
See if you can design your own model.

## SAMPLE RUN



## PROGRAM LISTING

```
100 PRINT "  
110 PRINT "  
120 PRINT "  
130 PRINT "  
140 PRINT "  
150 PRINT "  
160 PRINT "  
170 PRINT "  
180 PRINT "  
190 PRINT "  
200 PRINT "  
210 PRINT "  
220 PRINT "  
230 FOR I=1 TO 15  
240 FOR J=1 TO 100  
250 NEXT J  
260 PRINT  
270 NEXT I  
280 END
```



**Note:** This program shows you that you can put any number, letter or special character in a PRINT statement.

**Lines 230-270:** These statements make the rocket ship blast off your video terminal. They are explained further in Chapter 8.

*Special credit to Chris Carlson*

---

# KNOCK KNOCK

---

The computer can tell KNOCK KNOCK jokes. Here is one that we like. Can you write a program that tells some of your favorite KNOCK KNOCK jokes?

## SAMPLE RUN

KNOCK KNOCK

WHO'S THERE?

AMOS

AMOS WHO?

AMOS QUITO BIT ME!

## PROGRAM LISTING

```
100 PRINT "KNOCK KNOCK"  
110 PRINT  
120 PRINT "WHO'S THERE?"  
130 PRINT  
140 PRINT "AMOS"  
150 PRINT  
160 PRINT "AMOS WHO?"  
170 PRINT  
180 PRINT "AMOS QUITO BIT ME!"  
190 END
```

---

# RIDDLES

---

You can put riddles into a computer program too. Here are 3 riddles. See if you can figure out the answers.

Write a riddle program of your own using your favorite riddles.

## SAMPLE RUN

WHICH ANSWERS GO TO EACH OF THE FOLLWING RIDDLES?

1. WHY DID THE CHICKEN CROSS THE ROAD?
2. WHY DID THE FARMER CALL HIS PIG INK?
3. WHAT HAPPENS TO DUCKS WHEN THEY FLY UPSIDE DOWN?

## ANSWERS

- A. BECAUSE HE NEVER STAYED IN HIS PEN
- B. THEY QUACK UP
- C. TO GET TO THE OTHER SIDE

## PROGRAM LISTING

```
100 PRINT "WHICH ANSWERS GO TO EACH OF THE FOLLWING
RIDDLES?"
110 PRINT
120 PRINT "1. WHY DID THE CHICKEN CROSS THE ROAD?"
130 PRINT "2. WHY DID THE FARMER CALL HIS PIG INK?"
140 PRINT "3. WHAT HAPPENS TO DUCKS WHEN THEY FLY UPSIDE
DOWN?"
150 PRINT "      ----"
160 PRINT
170 PRINT "ANSWERS"
180 PRINT
190 PRINT "A. BECAUSE HE NEVER STAYED IN HIS PEN"
200 PRINT "B. THEY QUACK UP"
210 PRINT "C. TO GET TO THE OTHER SIDE"
220 END
```

*Special credit to Paul Zuanich*

---

# 2 LET

## Adding 2 and 2

---

You use this programming statement when you want to tell the computer to do arithmetic.

### PROBLEMS

1. Recycle
2. Baby Sister
3. Paper Route
4. Retirement
5. Moon
6. Pizza
7. Swim Team
8. Inheritance
9. Sleep-over

---

# RECYCLE

---

You and two of your friends are collecting aluminum cans. If you collect 200 cans, and one friend collects 140 cans and the other 325, how many will you have altogether? This program gives you the answer.

How would you change the program if you collected 300 cans?

If another friend wanted to join your group, how would you change the program in order to add in the cans this friend collected?

You can use this same program to add other things, like test scores, money you have earned or the number of records you own.

## SAMPLE RUN

200 + 140 + 325 = 665 TOTAL CANS COLLECTED

## PROGRAM LISTING

```
100 LET A=200
110 LET B=140
120 LET C=325
130 LET D=A+B+C
140 PRINT A;" + ";B;" + ";C;" = ";D;" TOTAL CANS
    COLLECTED"
150 END
```

Note: Lines 100-130: A, B, C and D are numeric variables, and they can be set equal to any number or calculation.

Line 130: The symbol + tells the computer to add.

## VARIABLE LIST

Tells you what the variables in a program are used for.

A—Number of cans you collected

B—Number of cans one friend collected

C—Number of cans the other friend collected

D—Sum of A, B, and C

## COMPUTER NOTES

Radio Shack TRS-80 Color BASIC Does not use the word LET. Just leave this word out when entering lines 100-130. For example:

Instead of	LET A = 200
Use	A = 200

Remember to leave out the word LET in all the programs in the rest of the text too.

---

# BABY SISTER

---

You are 10 years old now, and your baby sister is 2. How old will your sister be when you are 20? This program tells you.

How would you change the program if you were only 8 years old now?

What if your sister is older than you, can you still use this program?

## SAMPLE RUN

```
IF YOU AND YOUR SISTER ARE 8 YEARS APART IN AGE
YOUR SISTER WILL BE 12 WHEN YOU ARE 20
```

## PROGRAM LISTING

```
100 LET A=10
110 LET S=2
120 LET D=A-S
130 LET B=20-D
140 PRINT "IF YOU AND YOUR SISTER ARE ";D;" YEARS APART
    IN AGE"
150 PRINT "YOUR SISTER WILL BE ";B;" WHEN YOU ARE 20."
160 END
```

Note: Lines 120 and 130: The symbol - tells the computer to subtract.

## VARIABLE LIST

A—Your age now

S—Your sister's age now

D—Years difference

B—Your sister's age when you are 20

---

# PAPER ROUTE

---

You need to earn some money for Christmas presents. Your friend suggested that you deliver papers. If you expect to earn \$20 a week and there are  $2\frac{1}{2}$  months until Christmas, how much money will you have for presents? Use this program to get the answer.

Try changing the number of months or the amount of money you will earn each week.

## SAMPLE RUN

```
IF YOU EARN $ 20      EACH WEEK
YOU WILL HAVE $ 200   AFTER  2.5      MONTHS
```

## PROGRAM LISTING

```
100 LET W=20
110 LET M=2.5
120 LET T=4*M*W
130 PRINT "IF YOU EARN $";W;" EACH WEEK"
140 PRINT "YOU WILL HAVE $ ";T;" AFTER ";M;" MONTHS"
150 END
```

Note: Line 120: The symbol \* tells the computer to multiply.

Line 120: This program uses four weeks in a month to make the calculations easier.

## VARIABLE LIST

W—Amount of money you earn each week

M—Number of months until Christmas

T—Total amount of money you will earn

---

# RETIREMENT

---

If you want to earn \$900,000 before you retire and you can earn \$50,000 a year, how many years will you have to work before you can retire? This program gives you the answer.

Suppose you want to earn \$1,500,000. Can you change the program to give you the right answer?

## SAMPLE RUN

```
IF YOU EARN $ 50000.      PER YEAR
YOU WILL HAVE TO WORK FOR 18  YEARS
IN ORDER TO EARN $ 900000.
```

## PROGRAM LISTING

```
100 LET A=900000.
110 LET B=50000.
120 LET C=A/B
130 PRINT "IF YOU EARN $";B;" PER YEAR"
140 PRINT "YOU WILL HAVE TO WORK FOR ";C;" YEARS"
150 PRINT "IN ORDER TO EARN $ ";A
160 END
```

Note: Line 120: The symbol / tells the computer to divide.

## VARIABLE LIST

A—Total amount you want to earn  
B—Amount you can earn each year  
C—Number of years you must work

---

# MOON

---

If you weigh 50 kilograms on Earth, how much would you weigh on the moon? This program can calculate the answer.

What if you weigh 45 kilograms? How would you change the program?

## SAMPLE RUN

```
I CAN CONVERT YOUR EARTH WEIGHT TO YOUR MOON WEIGHT
IF YOU WEIGHT 50 KILOGRAMS ON EARTH
YOU WILL WEIGHT 8.33333 ON THE MOON
```

## PROGRAM LISTING

```
100 PRINT "I CAN CONVERT YOUR EARTH WEIGHT TO YOUR
      MOON WEIGHT"
110 C=1/6
120 W=50
130 M=W*C
140 PRINT "IF YOU WEIGH ";W;" KILOGRAMS ON EARTH"
150 PRINT "YOU WILL WEIGH ";M;" ON THE MOON"
160 END
```

Note: Lines 110-130: M, W and C are called numeric variables, and they can be set equal to any number.

## VARIABLE LIST

C—Conversion factor—Earth to moon weight

W—Your weight on Earth

M—Your weight on the moon

---

# PIZZA

---

Five of you are going out for pizza and cokes. You know that the pizza will cost \$10 and a coke will cost \$1. How much money should you take with you to pay for your share? Use this program for the answer.

Can you change the price of the pizza or the number of friends in this program?

What if you want 2 cokes? How would you change the program?

## SAMPLE RUN

```
TO PAY YOUR SHARE OF THE $,10 PIZZA
AND BUY A $ 1 COKE, YOU WILL NEED $ 3
```

## PROGRAM LISTING

```
100 LET P=10
110 LET C=1
120 LET F=5
130 LET M=P/F+C
140 PRINT "TO PAY YOUR SHARE OF THE $";P;" PIZZA"
150 PRINT "AND BUY A $";C;" COKE, YOU WILL NEED $";M
160 END
```

Note: Line 130: The computer divides first, then adds.

## VARIABLE LIST

P—Price of pizza  
C—Cost of coke  
F—Number of friends

---

# SWIM TEAM

---

In your workouts with the swim team, your coach has you swim 30 laps a day. If you swim 4 days a week, how many miles will you swim each week? This program gives you the answer for a pool 25 yards long.

Can you change the program so that you can find out how far you swim in a month?

What if your pool is only 20 yards long? How would you change the program?

## SAMPLE RUN

```
IF YOU SWIM 30 LAPS A DAY FOR 4 DAYS
YOU WILL SWIM A TOTAL OF 1.70455 MILES
```

## PROGRAM LISTING

```
100 LET L=30
110 LET D=4
120 LET N=L*D
130 LET Y=N*25
140 LET M=Y*3/5280
150 PRINT "IF YOU SWIM ";L;" LAPS A DAY FOR ";D;" DAYS"
160 PRINT "YOU WILL SWIM A TOTAL OF ";M;" MILES"
170 END
```

Note: Line 140: The computer multiplies first, then divides.

## VARIABLE LIST

L—Number of laps per day  
D—Number of days per week  
N—Total number of laps per week  
Y—Total number of yards per week  
M—Number of miles per week

---

# INHERITANCE

---

You have just received the news that you are to inherit some money from a distant uncle. You will receive \$200 a month until you are 18. If you are 10 now, how much money will you inherit altogether? Use this program to find out.

If you were to receive \$300 a month, how would you change this program? What if you are only 8 years old?

## SAMPLE RUN

```
IF YOU RECEIVE $ 200 FOR THE NEXT 96 MONTHS
YOU WILL RECEIVE $ 19200 IN TOTAL
```

## PROGRAM LISTING

```
100 LET A=200
110 LET Y=10
120 LET M=(18-Y)*12
130 LET T=M*A
140 PRINT "IF YOU RECEIVE $ ";A;" FOR THE NEXT
      ";M;" MONTHS"
150 PRINT "YOU WILL RECIEVE $";T;" IN TOTAL"
160 END
```

Note: Line 120: The ( ) tell the computer to subtract first, then multiply.

## VARIABLE LIST

A—Monthly payment

Y—Your current age

M—Number of months before you turn 18

T—Total amount you will receive

---

# SLEEP-OVER

---

You are planning a sleep-over with 5 friends. Your mother told you to buy 2 hot dogs, 3 candy bars and something to read for yourself and each guest. You also need some soda, and you know that 1 liter of soda is enough for 3 of you. How much food will you buy at the store? Use this program to get the answer.

What if one of your friends cannot eat candy? How would you change the program?

## SAMPLE RUN

```
FOR A SLEEP-OVER WITH 5 FRIENDS
YOU NEED 12 HOT DOGS, 18 CANDY BARS
2 LITERS OF SODA AND 6 COMIC BOOKS
```

## PROGRAM LISTING

```
100 LET A=5
110 LET B=2
120 LET C=3
130 LET D=3
150 LET E=(A+1)*B
160 LET F=(A+1)*C
170 LET G=(A+1)/D
180 PRINT "FOR A SLEEP-OVER WITH ";A;" FRIENDS"
190 PRINT "YOU NEED ";E;" HOT DOGS, ";F;" CANDY BARS"
200 PRINT G;" LITERS OF SODA AND ";A+1;" COMIC BOOKS"
210 END
```

Note: Lines 150, 160 and 170: The ( ) tell the computer to add first and then multiply or divide.

## VARIABLE LIST

A—Number of guests  
B—Number of hot dogs per person

- C—Number of candy bars per person
- D—Number of guests per 1 liter of soda
- E—Total number of hot dogs needed
- F—Total number of candy bars needed
- G—Total number of liters of soda needed

# 3

# GOTO

## Showing the Way

---

This programming statement allows you to tell the computer which line number to execute next.

This lets you repeat some lines in your program over and over.

### PROBLEMS

1. Stars
2. Counting
3. Power
4. Hunt
5. Gambler
6. Family Tree

---

# STARS

---

Write this program for your star-gazing friends. Hit the break key when you are tired of seeing stars.

Create your own universe or message by substituting your own lines.

## SAMPLE RUN

```
♦
♦♦
♦
♦
♦♦♦
♦
♦♦
♦ SEEING STARS?
♦♦
♦
♦♦♦
♦♦♦
```

## PROGRAM LISTING

```
100 REM♦♦THIS PROGRAM IS FOR STAR GAZERS♦♦
105 PRINT " ♦
110 PRINT "♦♦
115 PRINT " ♦
120 PRINT "♦
125 PRINT "
130 PRINT " ♦♦♦
135 PRINT " ♦♦♦♦
140 PRINT " ♦
145 PRINT " ♦♦
150 PRINT "
155 PRINT " ♦ SEEING STARS?♦
160 GOTO 105
165 END
```

**Note:** Line 100: REM is a "remark" statement that appears in the listing but will not print when the program is run. It's the programmer's way of storing information for future reference.

Line 160: GOTO 105 tells the computer to return to line 105 and run through the program again.

*Special credit to Felicia Lo.*

---

# COUNTING

---

This program counts by 2's. Hit your computer's break key to stop the program.

See if you can change it to count by 5's.

## SAMPLE RUN

I CAN COUNT BY 2

2

4

6

8

10

12

14

16

18

20

22

24

26

28

30

## PROGRAM LISTING

```
100 PRINT "I CAN COUNT BY 2"  
110 LET A=0  
115 LET A=A+2  
120 PRINT A  
125 GOTO 115  
130 END
```

Note: Line 110: Starts A at zero.

Line 115: Adds 2 more to A each time the computer returns to that line.

Line 125: Tells the computer to return to line 115 and run through the program again.

---

# POWER

---

This program prints all the powers of 2. Use your computer's break key to stop the counting.

Can you change it to print all the powers of 3 or 4?

## SAMPLE RUN

I CAN SHOW YOU ALL THE POWERS OF 2

2  
4  
8  
16  
32  
64  
128  
256  
512  
1024  
2048  
4096

## PROGRAM LISTING

```
100 PRINT "I CAN SHOW YOU ALL THE POWERS OF 2"  
110 LET B=2  
115 LET A=B  
120 PRINT A  
125 LET A=B+A  
130 GOTO 120  
135 END
```

Note: Can you tell what the computer is doing in line 125? Notice that the variable A is on both sides of the = sign.

## VARIABLE LIST

B—Number starting with  
A—Result of calculation

---

# HUNT

---

This program is a scrambled adventure. Thanks to the GOTO statement, the computer can unscramble and print out the adventure in a flash.

Can you beat the computer? Design a scrambled program of your own for your friends to try.

## SAMPLE RUN

YOU ARE ON AN EXPEDITION SEARCHING FOR THE LOST ARK.  
A FRIEND SHOWS YOU A LONG LOST MAP AND 'X' MARKS THE  
SPOT

WHERE IT WAS BURIED.

YOU SEE THE 'X' ON THE MAP. IT IS BY THE TEMPLE OF THE  
ANCIENT SUN GOD.

YOUR CREW STARTS DIGGING.

THEY UNCOVER A PIT FULL OF SNAKES, BUT THE ARK  
IS THERE!!!

YOU ENTER THE PIT WITH ROPES, AVOIDING THE SNAKES,  
AND SECURE THE ROPES TO THE ARK. A HELICOPTER SAFELY  
HOISTS IT OUT, BUT THERE ARE NO ROPES LEFT FOR YOU.  
TOUGH LUCK!!! THINK OF YOUR OWN SOLUTION.

## PROGRAM LISTING

```
100 PRINT "YOU ARE ON AN EXPEDITION SEARCHING FOR THE  
    LOST ARK."  
110 GOTO 260  
120 PRINT "YOUR CREW STARTS DIGGING."  
130 GOTO 220  
140 PRINT "AND SECURE THE ROPES TO THE ARK. A HELICOPTER  
    SAFELY"  
150 GOTO 280  
160 PRINT "WHERE IT WAS BURIED."  
170 GOTO 240  
180 PRINT "TOUGH LUCK!!! THINK OF YOUR OWN SOLUTION."  
190 GOTO 340  
200 PRINT "IS THERE!!!"  
210 GOTO 300  
220 PRINT "THEY UNCOVER A PIT FULL OF SNAKES, BUT  
    THE ARK"  
230 GOTO 200
```

```
240 PRINT "YOU SEE THE 'X' ON THE MAP. IT IS BY THE  
    TEMPLE OF THE"  
250 GOTO 320  
260 PRINT "A FRIEND SHOWS YOU A LONG LOST MAP AND 'X'  
    MARKS THE SPOT"  
270 GOTO 160  
280 PRINT "HOISTS IT OUT, BUT THERE ARE NO ROPES LEFT  
    FOR YOU."  
290 GOTO 180  
300 PRINT "YOU ENTER THE PIT WITH ROPES, AVOIDING THE  
    SNAKES,"  
310 GOTO 140  
320 PRINT "ANCIENT SUN GOD."  
330 GOTO 120  
340 END
```

---

# GAMBLER

---

If you bet \$2 at the blackjack tables at 2 to 1 odds and let your winnings ride, how quickly would your money grow? This program tells you. Use your computer's break key to stop the program.

How would you change the program if you wanted to start with a \$5 bet?

## SAMPLE RUN

THIS PROGRAM SHOWS YOU HOW A \$2 BET AT THE BLACKJACK TABLE WILL GROW IF YOU KEEP WINNING AND LET YOUR WINNINGS RIDE

ROUND	WINNINGS
1	\$ 5
2	\$ 12.5
3	\$ 31.25
4	\$ 78.13
5	\$ 195.31
6	\$ 488.28
7	\$ 1220.7
8	\$ 3051.76

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM SHOWS YOU HOW A $2 BET
    AT THE BLACKJACK TABLE"
105 PRINT "WILL GROW IF YOU KEEP WINNING AND
    LET YOUR WINNINGS RIDE"
110 PRINT
115 LET A=2
120 LET B=0
125 PRINT "      ROUND                WINNINGS"
130 LET A=A+A+A/2
135 LET B=B+1
140 LET C=INT(A*100+.5)/100
145 PRINT "      ";B;"                $" ;C
150 GOTO 130
155 END
```

**Note: Line 140:** This formula rounds your winnings to the nearest cent.

**Line 145:** A ; or a ; is required in a PRINT STATEMENT between a numeric variable and the quotation marks. A ; uses fewer spaces than a , . However, the spacing varies among computers so you may have to experiment with yours.

---

# FAMILY TREE

---

This program demonstrates how the number of people in your family can increase. Use your computer's break key to stop the program.

See what happens if you change the number of children each person has, or the time between generations.

Write a similar program to estimate the number of offspring your pet could have.

## SAMPLE RUN

THIS PROGRAM SHOWS HOW THE NUMBER OF PEOPLE IN YOUR  
FAMILY  
CAN INCREASE

STARTING WITH 2 DESCENDENTS  
IF EACH PERSON HAS 2 CHILDREN  
WITH 20 YEARS BETWEEN GENERATIONS

YEARS	DESCENDENTS
20	4
40	8
60	16
80	32
100	64
120	128
140	256
160	512
180	1024
200	2048
220	4096
240	8192
260	16384

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM SHOWS HOW THE NUMBER OF PEOPLE  
    IN YOUR FAMILY "  
105 PRINT "CAN INCREASE"  
115 PRINT  
120 T=20  
125 Y=0
```

```
130 A=2
135 C=2
140 D=A
145 PRINT "STARTING WITH ";A;" DESCENDENTS"
150 PRINT "IF EACH PERSON HAS ";C;" CHILDREN"
155 PRINT "WITH ";T;" YEARS BETWEEN GENERATIONS"
160 PRINT
165 PRINT "   YEARS           DESCENDENTS"
170 D=D+C
175 A=A+D
180 Y=Y+T
185 PRINT "   ";Y;"           ";D
190 GOTO 170
195 END
```

### VARIABLE LIST

T—Years between generations  
A—Number of descendants in first generation  
C—Number of children per person

# 4

# INPUT

## Talking to the Computer

---

This programming statement allows you to enter different numbers into your program for the same problem each time it is run.

### PROBLEMS

1. Conversion
2. Silicon Valley Game
3. The Great Magician
4. Batting Averages
5. Year 2000
6. Averages
7. Savings Account
8. Mileage
9. Recipe

---

# CONVERSION

---

Do you know how many centimeters tall you are? Do you know how many centimeters tall your father is? You can use this program to convert inches to centimeters.

You could also use a very similar program to convert pounds to kilograms or gallons to liters.

To stop this program, hit the break key instead of entering a number.

## SAMPLE RUN

CONVERT INCHES TO CENTIMETERS

INCHES?60  
152.4           CENTIMETERS

INCHES?100  
254   CENTIMETERS

## PROGRAM LISTING

```
100 PRINT "CONVERT INCHES TO CENTIMETERS"  
120 PRINT  
130 PRINT "INCHES";  
140 INPUT I  
150 LET C=I*2.54  
160 PRINT C;" CENTIMETERS"  
170 GOTO 120  
180 END
```

Note: Line 140: The INPUT statement allows you to type in different numeric values for I each time this statement is run.

## VARIABLE LIST

I—Inches  
C—Centimeters

---

# SILICON VALLEY GAME

---

If you were a Silicon Valley executive and you had \$200,000 to spend, how many Mercedes could you buy? You can use this program to get the answer. To stop the program, hit the break key instead of entering a number.

What if you decided to economize and only spend \$100,000, how would you change this program?

## SAMPLE RUN

```
YOU ARE A SILICON VALLEY EXECUTIVE AND  
YOU HAVE 200,000 DOLLARS TO SPEND ON MERCEDES  
HOW MANY CAN YOU BUY?
```

```
TYPE IN THE PRICE OF ONE MERCEDES?30000  
YOU COULD BUY 6 MERCEDES  
AND HAVE 20000 DOLLARS LEFT OVER.
```

## PROGRAM LISTING

```
100 PRINT "YOU ARE A SILICON VALLEY EXECUTIVE AND"  
110 PRINT "YOU HAVE 200,000 DOLLARS TO SPEND ON  
MERCEDES"  
120 PRINT "HOW MANY CAN YOU BUY?"  
140 PRINT  
150 PRINT "TYPE IN THE PRICE OF ONE MERCEDES";  
160 INPUT A  
170 LET B=200000./A  
180 LET B=INT(B)  
190 LET T=200000.-B*A  
200 PRINT "YOU COULD BUY ";B;" MERCEDES"  
210 PRINT "AND HAVE ";T;" DOLLARS LEFT OVER."  
220 GOTO 140  
230 END
```

Note: Line 150: The ; stops the computer from moving to the next line. This keeps the input on the same line as the print.

Line 180: This statement forces B to be a whole number. Chapter 7 explains how the INT (Integer) function works.

## VARIABLE LIST

A—Price of one Mercedes

B—Number of Mercedes you could buy

T—Money you have left over

---

# THE GREAT MAGICIAN

---

Teach the computer a magic trick! Think of a number, follow the instructions and this program will guess your number. Don't be surprised if the computer is right every time.

Can you figure out how it does the trick? Can you write a magic trick of your own?

## SAMPLE RUN

```
HI, I AM THE GREAT MAGICIAN.  
THINK OF A NUMBER, ANSWER MY QUESTION  
AND I WILL TRY TO GUESS YOUR NUMBER.  
OK, THINK OF A NUMBER- BUT DON'T TELL ME YET.
```

```
NOW, TAKE YOUR NUMBER AND ADD 5.  
MULTIPLY THE RESULT BY 3 AND THEN SUBTRACT 8.  
MULTIPLY THIS RESULT BY 2 AND THEN ADD 4.  
WHAT DO YOU HAVE??  
YOUR NUMBER IS 10
```

## PROGRAM LISTING

```
100 REM◆◆NUMBER GUESSING GAME◆◆  
110 PRINT "HI, I AM THE GREAT MAGICIAN."  
120 PRINT "THINK OF A NUMBER, ANSWER MY QUESTION"  
130 PRINT "AND I WILL TRY TO GUESS YOUR NUMBER."  
140 PRINT "OK, THINK OF A NUMBER- BUT DON'T TELL ME YET."  
150 PRINT  
160 PRINT "NOW TAKE YOUR NUMBER AND ADD 3. DIVIDE THE  
RESULT BY 5."  
170 PRINT "NOW MULTIPLY BY 8, DIVIDE BY 5 AND ADD 5.  
SUBTRACT 1."  
180 PRINT "WHAT DO YOU HAVE?";  
190 INPUT B  
200 LET C=(B+1-5)◆5/8◆5-3  
210 PRINT "YOUR NUMBER IS ";C  
220 END
```

Note: Line 200: The computer does calculations with numeric variables. This is called a numeric expression.

## VARIABLE LIST

**B—Result of calculations**

**C—Number to guess**

---

# BATTING AVERAGES

---

This program calculates batting averages. You could use it to help your coach pick the players for your All Star team.

## SAMPLE RUN

CALCULATE A PLAYER'S BATTING AVERAGE

NUMBER OF TIMES AT BAT?30  
NUMBER OF HITS?6  
NUMBER OF WALKS?12  
YOUR BATTING AVERAGE IS .333333

NUMBER OF TIMES AT BAT?15  
NUMBER OF HITS?5  
NUMBER OF WALKS?0  
YOUR BATTING AVERAGE IS .333333

## PROGRAM LISTING

```
100 PRINT "CALCULATE A PLAYER'S BATTING AVERAGE"  
120 PRINT  
130 PRINT "NUMBER OF TIMES AT BAT";  
140 INPUT B  
150 PRINT "NUMBER OF HITS";  
160 INPUT H  
170 PRINT "NUMBER OF WALKS";  
180 INPUT W  
190 LET A=H/(B-W)  
200 PRINT "YOUR BATTING AVERAGE IS ";A  
210 PRINT  
220 GOTO 130  
230 END
```

Note: Notice that this program uses 3 different input statements.

## VARIABLE LIST

B—Number of times at bat  
H—Number of hits  
W—Number of walks  
A—Batting average

---

# YEAR 2000

---

How old will you be in the year 2000? How old will your best friend be? You can use this program to find out.

How would you change this program if you wanted to know your age in the year 2025?

## SAMPLE RUN

FIND OUT HOW OLD YOU WILL BE IN THE YEAR 2000

HOW OLD ARE YOU NOW?12  
WHAT YEAR IS IT NOW?1982  
IN THE YEAR 2000, YOU WILL BE 30

HOW OLD ARE YOU NOW?35  
WHAT YEAR IS IT NOW?1982  
IN THE YEAR 2000, YOU WILL BE 53

## PROGRAM LISTING

```
100 PRINT "FIND OUT HOW OLD YOU WILL BE IN THE YEAR  
2000"  
120 PRINT  
130 PRINT "HOW OLD ARE YOU NOW";  
140 INPUT A  
150 PRINT "WHAT YEAR IS IT NOW";  
160 INPUT Y  
170 REM◆◆NOW CALCULATE THE NUMBER OF YEARS  
180 LET T=2000-Y+A  
190 PRINT "IN THE YEAR 2000, YOU WILL BE ";T  
200 GOTO 120  
210 END
```

## VARIABLE LIST

A—Your age now

Y—Current year

T—Your age in the year 2000

---

# AVERAGES

---

This program calculates the average of 5 numbers. You can use it to calculate test averages, the average age of 5 of your friends, or the average temperature over 5 days. Can you think of other uses for this program?

What if you need the average of 6 numbers? Can you change this program to calculate your answers?

## SAMPLE RUN

```
CALCULATE THE AVERAGE OF FIVE NUMBERS
INPUT THE NUMBERS LIKE THIS 1,2,3,4,5
```

```
NUMBERS?10,20,30,40,50
AVERAGE IS 30
```

## PROGRAM LISTING

```
100 PRINT "CALCULATE THE AVERAGE OF FIVE NUMBERS"
110 PRINT "INPUT THE NUMBERS LIKE THIS 1,2,3,4,5"
120 PRINT
130 PRINT "NUMBERS";
140 INPUT A,B,C,D,E
150 LET T=(A+B+C+D+E)/5
160 PRINT "AVERAGE IS ";T
170 PRINT
180 END
```

Note: Line 150: The ( ) tell the computer to add all 5 numbers together before dividing.

Line 140: Notice that you can put in more than 1 number in an INPUT statement.

## VARIABLE LIST

T—Average of 5 numbers

---

# SAVINGS ACCOUNT

---

Suppose your grandmother gives you some money to save for your college education. If you put it in the bank at the current interest rate, how much money will you have when you are ready to start college?

You can change the interest rate or the original amount, and see how much money you have.

## SAMPLE RUN

FIND OUT HOW YOUR MONEY WILL GROW IN YOUR SAVINGS ACCOUNT

ORIGINAL AMOUNT?100  
NUMBER OF YEARS?5  
YEARLY INTEREST RATE (IN DECIMAL)?.055  
AFTER 5 YEARS, YOU WILL HAVE 130.7 DOLLARS

ORIGINAL AMOUNT?200  
NUMBER OF YEARS?10  
YEARLY INTEREST RATE (IN DECIMAL)?.15  
AFTER 10 YEARS, YOU WILL HAVE 809.11 DOLLARS

## PROGRAM LISTING

```
100 PRINT "FIND OUT HOW YOUR MONEY WILL GROW IN YOUR
    SAVINGS ACCOUNT"
120 PRINT
130 PRINT "ORIGINAL AMOUNT";
140 INPUT A
150 PRINT "NUMBER OF YEARS";
160 INPUT N
170 PRINT "YEARLY INTEREST RATE (IN DECIMAL)";
180 INPUT I
190 LET T=A*(1+I) ** N
200 LET T=INT(T*100+.5)/100
210 PRINT "AFTER ";N;" YEARS, YOU WILL HAVE ";T;"
    DOLLARS"
220 PRINT
230 GOTO 120
240 END
```

Note: Line 190: The symbol **\*\*** tells the computer to raise a number to a power, ie. **LETX = 2\*\*4** tells the computer to raise 2 to the 4th power. Please note—your computer may use a different symbol. Refer to your BASIC manual under exponentiation to find out what your computer uses.

Line 200: This statement rounds the dollar amount to the nearest cent.

### VARIABLE LIST

A—Original amount of money

N—Number of years to save

I—Yearly interest rate

T—Amount of money you have for college

---

# MILEAGE

---

You are planning on taking a driving vacation with your parents. Your father wants to know how long it will take to get to your destination and how much money gas will cost. You can use this program to get the answers.

Can you add to the program to include food and lodging expenses for your trip?

What would you change if you wanted to use this program for a trip on a plane or train?

## SAMPLE RUN

TAKING A VACATION ? USE THIS PROGRAM TO  
CALCULATE YOUR DRIVING TIME AND  
COST OF GASOLINE

MILES TO TRAVEL?500  
MILES PER GALLON?15  
PRICE PER GALLON?1.30  
MILES PER HOUR?55  
IT WILL TAKE 9.09091 HOURS AND COST 43.3333  
DOLLARS.

## PROGRAM LISTING

```
100 PRINT "TAKING A VACATION ? USE THIS PROGRAM TO"  
110 PRINT "CALCULATE YOUR DRIVING TIME AND"  
120 PRINT "COST OF GASOLINE"  
130 PRINT  
140 PRINT "MILES TO TRAVEL";  
150 INPUT M  
160 PRINT "MILES PER GALLON";  
170 INPUT A  
180 PRINT "PRICE PER GALLON";  
190 INPUT P  
200 PRINT "MILES PER HOUR";  
210 INPUT S  
220 LET T=M/S  
230 LET C=(M/A)*P  
240 PRINT "IT WILL TAKE ";T;" HOURS AND COST ";C;"  
DOLLARS."  
250 END
```

## VARIABLE LIST

M—Miles to travel

A—Miles per gallon

P—Price per gallon

S—Miles per hour

T—Number of hours to reach destination

C—Total cost of gasoline

---

# RECIPE

---

The computer can also help you out in the kitchen. Suppose you need to increase the size of your favorite recipe in order to feed a large crowd. This program converts the amount of ingredients you will need.

Use this program to make a recipe smaller also.

## SAMPLE RUN

THIS PROGRAM ALTERS YOUR FAVORITE RECIPE.

NUMBER THE RECIPE ORIGINALLY SERVED?4

NUMBER NEW RECIPE TO SERVE?8

NOW INPUT THE ORIGINAL QUANTITY OF EACH  
INGREDIENT (IN DECIMAL)?2

NEW QUANTITY 4

INGREDIENT (IN DECIMAL)?.5

NEW QUANTITY 1

INGREDIENT (IN DECIMAL)?1.4

NEW QUANTITY 2.8

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM ALTERS YOUR FAVORITE RECIPE."  
110 PRINT "NUMBER THE RECIPE ORIGINALLY SERVED";  
120 INPUT F  
130 PRINT "NUMBER NEW RECIPE TO SERVE";  
140 INPUT N  
150 PRINT "NOW INPUT THE ORIGINAL QUANTITY OF EACH"  
160 PRINT "INGREDIENT (IN DECIMAL)";  
170 INPUT I  
200 PRINT "NEW QUANTITY "; I*(N/F)  
210 PRINT  
220 GOTO 160  
230 END
```

Note: Line 200: A numeric expression can be used in a PRINT statement. The computer just prints the result of the calculation.

## VARIABLE LIST

F—Number of people recipe originally served

N—Number of people the new recipe is to serve

I—Original quantity of an ingredient

---

# 5 STRINGS

## Using Words with the Computer

---

A string variable contains letters or words instead of numbers. You can print strings, combine them to form words or sentences and input strings just as you can numbers.

### PROBLEMS

1. Nonsense
2. A Man From Peru
3. A Young Lady Named Maude
4. Race
5. Rhyme
6. Names
7. Author

---

# NONSENSE

---

This program prints out a message. Can you see where each word is stored in the program?

Can you write a scrambled message of your own? Which lines in the program must you change?

## SAMPLE RUN

DID YOU THINK THIS WOULD PRINT NONSENSE?

## PROGRAM LISTING

```
100 DIM A$(10),B$(10),C$(10),D$(10),E$(10),F$(10)
110 LET A$="SENSE?"
120 LET B$="DID YOU TH"
130 LET C$="INK THIS "
140 LET D$="INT NON"
150 LET E$="LD PR"
160 LET F$="WOU"
170 PRINT B$;C$;F$;E$;D$;A$
180 END
```

Note: Line 100: This is a dimension (DIM) statement. If you want a string variable to have more than one letter in it, you must use the DIM statement to save the spaces. In this case, A\$ can have 10 letters, B\$ can have 10 letters, etc.

Lines 110-160: A\$, B\$, etc., are used to store letters in the program. These variables are called string variables. The \$ tells the computer this is a string variable.

## COMPUTER NOTES

TI-BASIC Do not use the DIM statement for string variables; the computer saves space automatically. Therefore, take out the string variables in the DIM statements in all the programs in this text. See Appendix A for a more detailed explanation.

---

# A MAN FROM PERU

---

You can write a limerick like this one yourself. Just substitute your own words for A\$, B\$ and C\$.

If you have another limerick you like better, try putting it on the computer.

## SAMPLE RUN

```
THERE WAS AN OLD MAN FROM PERU
WHO DREAMED HE WAS EATING HIS SHOE
HE WOKE UP IN THE NIGHT, WITH A TERRIBLE FRIGHT
TO FIND OUT IT WAS PERFECTLY TRUE.
```

## PROGRAM LISTING

```
100 DIM A$(10),B$(10),C$(10)
110 LET A$="PERU"
120 LET B$="SHOE"
130 LET C$="TRUE"
140 PRINT "THERE WAS AN OLD MAN FROM ";A$
150 PRINT "WHO DREAMED HE WAS EATING HIS ";B$
160 PRINT "HE WOKE UP IN THE NIGHT, WITH A TERRIBLE
    FRIGHT"
170 PRINT "TO FIND OUT IT WAS PERFECTLY ";C$;"."
180 END
```

Note: Lines 140-170: You can use the string variables A\$, B\$ and C\$ in a PRINT statement just as you used the numeric variables.

---

# A YOUNG LADY NAMED MAUDE

---

Here is another limerick. Input your own strings for your original version. It is easier to do it with this program, because the program asks you for each word.

## SAMPLE RUN

```
THIS PROGRAM WRITES A LIMERICK
INPUT 3 WORDS THAT RHYME
FIRST WORD?MAUDE
SECOND WORD?BROAD
THIRD WORD?O LORD
```

```
THERE WAS A YOUNG LADY NAMED MAUDE
WHOSE FRAME WAS EXCEEDINGLY BROAD
TO EAT AT THE TABLE, SHE SCARCELY WAS ABLE
BUT OUT IN THE PANTRY, O LORD
```

## PROGRAM LISTING

```
100 DIM A$(25),B$(25),C$(25)
110 PRINT "THIS PROGRAM WRITES A LIMERICK"
120 PRINT "INPUT 3 WORDS THAT RHYME"
130 PRINT "FIRST WORD";
140 INPUT A$
150 PRINT "SECOND WORD";
160 INPUT B$
170 PRINT "THIRD WORD";
180 INPUT C$
190 PRINT
200 PRINT
210 PRINT "THERE WAS A YOUNG LADY NAMED ";A$
220 PRINT "WHOSE FRAME WAS EXCEEDINGLY ";B$
230 PRINT "TO EAT AT THE TABLE, SHE SCARCELY WAS ABLE"
240 PRINT "BUT OUT IN THE PANTRY, ";C$
250 END
```

Note: Lines 140, 160, 180: The INPUT statement works for string variables as well as for numeric variables.

---

# RACE

---

This program tells the story of a crazy race using names, a noun and an adjective that you choose.

Can you change the program to broadcast a game or tell a story?

## SAMPLE RUN

WELCOME TO THE SCRAMBLED SWEEPSTAKES!

I NEED A PLURAL NOUN?HUNCHBACKS  
I NEED A NAME?TIM CONWAY  
I NEED A SECOND NAME?STEVE MARTIN  
I NEED A THIRD NAME?ALAN ALDA  
I NEED AN ADJECTIVE?FANTASTIC

ALL RIGHT, THE RACE IS ABOUT TO BEGIN!  
THIS IS YOUR ANNOUNCER SPEAKING.  
THE HUNCHBACKS ARE AT THE STARTING GATE...  
AND THEY'RE OFF!

STEVE MARTIN IS IN THE LEAD. ALAN ALDA IS NEXT, TRAILED  
BY TIM CONWAY.  
THE HUNCHBACKS ARE COMING INTO THE STRETCH.  
ALAN ALDA IS AHEAD AND STEVE MARTIN AND TIM CONWAY ARE  
MOVING UP.  
IT'S GOING TO BE CLOSE FOLKS... AND...  
STEVE MARTIN CROSSES THE WIRE FIRST!!!

WHAT A FANTASTIC DAY AT THE TRACK!

## PROGRAM LISTING

```
150 DIM A$(25),B$(25),C$(25),D$(25),E$(25)
160 PRINT
170 PRINT "WELCOME TO THE SCRAMBLED SWEEPSTAKES!"
180 PRINT
200 PRINT "I NEED A PLURAL NOUN";
205 INPUT A$
210 PRINT "I NEED A NAME";
220 INPUT B$
230 PRINT "I NEED A SECOND NAME";
240 INPUT C$
250 PRINT "I NEED A THIRD NAME";
```

```

260 INPUT D$
350 PRINT "I NEED AN ADJECTIVE";
360 INPUT E$
370 PRINT
380 PRINT "ALL RIGHT, THE RACE IS ABOUT TO BEGIN!"
390 PRINT "THIS IS YOUR ANNOUNCER SPEAKING."
400 PRINT "THE ";A$;" ARE AT THE STARTING GATE..."
410 PRINT "AND THEY'RE OFF!"
420 PRINT
430 PRINT C$;" IS IN THE LEAD. ";D$;" IS NEXT, TRAILED"
440 PRINT "BY ";B$;". "
450 PRINT "THE ";A$;" ARE COMING INTO THE STRETCH."
460 PRINT D$;" IS AHEAD AND ";C$;" AND ";B$;"
   ARE MOVING UP."
465 PRINT "IT'S GOING TO BE CLOSE FOLKS... AND..."
470 PRINT C$;" CROSSES THE WIRE FIRST!!!"
480 PRINT
490 PRINT "WHAT A ";E$;" DAY AT THE TRACK!"
500 END

```

#### VARIABLE LIST

A\$—Plural noun  
B\$—First name  
C\$—Second name  
D\$—Third name  
E\$—Adjective

---

# RHYME

---

Your teacher told you to write a poem, and you are struggling with words to rhyme with your lines. Use this program to test out various possibilities.

Can you change it to handle words starting with digraphs(th,wh) or blends(st,tr)?

## SAMPLE RUN

TEST RHYMING POSSIBILITIES WITH THIS PROGRAM

WHAT WORD ARE YOU RHYMING?

?CAT

NEW FIRST LETTER?H

HAT

NEW FIRST LETTER?B

BAT

NEW FIRST LETTER?S

SAT

## PROGRAM LISTING

```
100 DIM Z$(10),X$(10),D$(1)
110 PRINT "TEST RHYMING POSSIBILITIES WITH THIS
PROGRAM"
120 PRINT "WHAT WORD ARE YOU RHYMING?"
130 INPUT Z$
140 X$ = MID$(Z$,2)
145 PRINT "NEW FIRST LETTER";
150 INPUT D$
170 PRINT D$;X$
175 PRINT
180 GOTO 145
190 END
```

Note: Line 140: This statement puts Z\$ into X\$ starting with the second letter of Z\$.

The string function MID\$ allows you to take out part of a string. The format is MID\$(string,n,m) where:

string—the name of the string variable  
n—the position of the first letter to use  
m—number of letters to use, m is not required if you  
want to use the entire string.

### VARIABLE LIST

Z\$—Contains word trying to rhyme  
D\$—Contains the letter input to make a new word  
X\$—Contains the letters in Z\$ starting with the second letter

### COMPUTER NOTES

HP-2000 BASIC	}	Replace line 140 with:
Atari BASIC		140 LET X\$ = Z\$(2)
TI-BASIC		Not suitable for the TI. See Appendix A.

---

# NAMES

---

This program makes up a child's name by combining a first name and a last name. What names can you come up with?

Use a similar program to make up names for your soccer team or club.

Try changing the program so you can use it for compound words.

## SAMPLE RUN

```
USE THIS PROGRAM TO EXPERIMENT WITH NAMES  
IF PARENT'S WITH A LAST NAME OF ?WAHL
```

```
NAMED THEIR CHILD ?BRICK  
THE CHILD'S FULL NAME WOULD BE BRICK WAHL
```

## PROGRAM LISTING

```
100 DIM A$(15),B$(15)  
110 PRINT "USE THIS PROGRAM TO EXPERIMENT WITH NAMES"  
120 PRINT "IF PARENTS WITH A LAST NAME OF ";  
130 INPUT A$  
140 PRINT  
150 PRINT "NAMED THEIR CHILD ";  
160 INPUT B$  
170 PRINT "THE CHILD'S FULL NAME WOULD BE ";B$;" ";A$  
180 GOTO 140  
190 END
```

## VARIABLE LIST

A\$—Parents' last name  
B\$—Child's first name

---

# AUTHOR

---

Change the words in a sentence to alter the message or improve your writing style. Change line 110 if you want to start with a different message.

You can only have 30 characters in variable A\$. What happens if you try to use more than 30? How would you change the program if you wanted more than 30?

## SAMPLE RUN

```
SHERRI IS A PRETTY GIRL
WHAT WORD DO YOU WANT TO CHANGE?SHERRI
WITH WHAT?LYN
LYN IS A PRETTY GIRL
WHAT WORD DO YOU WANT TO CHANGE?PRETTY
WITH WHAT?SMART
LYN IS A SMART GIRL
WHAT WORD DO YOU WANT TO CHANGE?GIRL
WITH WHAT?PERSON
LYN IS A SMART PERSON
```

## PROGRAM LISTING

```
100 DIM A$(30),B$(39),C$(10),D$(10)
110 LET A$ = " SHERRI IS A PRETTY GIRL"
120 PRINT A$
130 PRINT "WHAT WORD DO YOU WANT TO CHANGE";
140 INPUT C$
150 REM ** CREATE A NEW STRING
160 PRINT "WITH WHAT";
170 INPUT D$
180 REM **FIND FIRST POSITION IN A$ OF C$
182 LET F = LEN (C$)
185 LET A = LEN (A$)
190 FOR B = 1 TO A
200 IF MID$(A$,B,F) = C$ THEN 240
210 NEXT B
220 PRINT C$" NOT IN",A$,"TRY AGAIN"
230 GOTO 130
240 LET B$ = A$
250 A$ = LEFT$(B$,B - 1) + D$ + MID$(B$,B + F)
260 GOTO 120
270 END
```

**Note:** Line 182: The LEN (length) statement determines the actual length of the string in A\$. In this case, A is set equal to the actual length of A\$.

Line 200: This statement determines the starting position of the word C\$ in the sentence A\$.

Line 250: This statement changes A\$ by replacing the old word C\$ with the new word D\$.

The string function LEFT\$ allows you to use only the first part of a string variable (the left-most letters). The format is LEFT\$(string, n) where:

string—name of the string variable

n—number of letters to use.

## VARIABLE LIST

A\$—Expression starting with

B\$—Temporary variable used to hold expression in A\$ while changes are made

C\$—Word replacing

D\$—Word adding

A—Length of A\$

F—Length of C\$

B—Starting position of C\$ in A\$

## COMPUTER NOTES

HP-2000 BASIC

Replace lines 190-210 with:

190 LET B=POS(A\$,C\$)

Replace line 250 with:

250 LET A\$(B,B+D) = D\$

255 LET A\$(B+D+1) = B\$(B+F+1)

The POS (position) function determines the starting position of the word C\$ in the sentence A\$. The format is POS(string1,string2) where:

string1—name of string containing string2

Atari BASIC }  
TI-BASIC }

Not suitable for use with these versions of BASIC

---

# 6 IF . . . THEN

## How the Computer Makes Decisions

---

This programming statement makes the computer decide which line number to execute next.

### PROBLEMS

1. Tax
2. Foreign Currency
3. Mystery Code
4. Ten Speed
5. Time Differences
6. Interior Decorating
7. Driver's Test
8. Spelling
9. You Are a Detective

---

# TAX

---

Sales tax is added to the price of most things you buy. In order to be sure you take enough money with you, use this program to find out exactly how much money you need to buy that rocket you've been looking at. Or, any other item you are thinking of buying. Your parents may find it useful too.

This program uses a tax of .065¢. If you live in a different state, the tax might be different. Can you change the program for a different tax rate?

## SAMPLE RUN

I WILL FIGURE OUT HOW MUCH TAX WILL BE CHARGED ON AN ITEM AND ADD IT TO YOUR PURCHASE. JUST INPUT THE PRICE.  
INPUT '0' TO STOP.

PRICE OF ITEM?4.98  
YOUR ITEM WILL COST \$ 5.3  
PRICE OF ITEM?16.95  
YOUR ITEM WILL COST \$ 18.05  
PRICE OF ITEM?32.25  
YOUR ITEM WILL COST \$ 34.35

## PROGRAM LISTING

```
100 PRINT "I WILL FIGURE OUT HOW MUCH TAX WILL BE
    CHARGED ON AN ITEM"
110 PRINT "AND ADD IT TO YOUR PURCHASE. JUST INPUT
    THE PRICE."
120 PRINT "INPUT '0' TO STOP."
130 PRINT
140 PRINT "PRICE OF ITEM";
150 INPUT P
160 IF P=0 THEN 200
170 LET P=P+.065*P
180 PRINT "YOUR ITEM WILL COST $";INT(P*100+.5)/100
190 GOTO 140
200 END
```

**Note:** Line 160: This program uses another way to stop. If the input is equal to zero, then the program skips to line 200 and ends.

**VARIABLE LIST**

**P—Price of an item**

*Special credit to Slaton Lipscomb*

---

# FOREIGN CURRENCY

---

Are you planning a vacation to a foreign country? You may have to change U.S. dollars into foreign currency. Do you know what the exchange rate will be? If you do, you can use this program to convert the U.S. dollar amount to foreign currency.

Can you write a program that converts foreign currency into U.S. dollars?

## SAMPLE RUN

THIS PROGRAM CAN CONVERT OUR DOLLAR TO ANY FOREIGN  
CURRENCY

INPUT A 0 TO STOP PROGRAM

INPUT THE FOREIGN EQUIVALENT OF A DOLLAR?.9982

INPUT THE DOLLAR AMOUNT?100

100 U.S. DOLLARS EQUAL 99.82 IN FOREIGN CURRENCY

INPUT THE DOLLAR AMOUNT?2500

2500 U.S. DOLLARS EQUAL 2495.5 IN FOREIGN CURRENCY

INPUT THE DOLLAR AMOUNT?53

53 U.S. DOLLARS EQUAL 52.9 IN FOREIGN CURRENCY

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM CAN CONVERT OUR DOLLAR TO ANY
    FOREIGN CURRENCY"
105 PRINT "INPUT A 0 TO STOP PROGRAM"
110 PRINT
115 PRINT "INPUT THE FOREIGN EQUIVALENT OF A DOLLAR";
120 INPUT F
125 PRINT "INPUT THE DOLLAR AMOUNT";
130 INPUT D
135 IF D=0 THEN 160
140 LET N=F*D
145 PRINT D;"U.S. DOLLARS EQUAL ";INT(N*100+.5)/100;"
    IN FOREIGN CURRENCY"
150 PRINT
155 GOTO 125
160 END
```



---

# MYSTERY CODE

---

In order to finish this program, you have to decipher a secret message while the program is running.

Can you write a mystery code program like this one and let your friends try to solve it?

## SAMPLE RUN

```
YOU ARE A BRITISH DIPLOMAT BEING HELD CAPTIVE BY A
TERRORIST GROUP.
SOMEONE JUST SLIPPED YOU A SECRET MESSAGE.
DECODE IT AND YOU CAN ESCAPE. HURRY THE TERRORISTS
ARE RETURNING!
HINT-CROSS OUT WORDS THAT RHYME
THE CODE IS.....
```

```
PSATRMATEFATSPATSCATLBATERATTYATTHATEZATRDATA
?A
KEY IS UNDER THE BOOK
```

## PROGRAM LISTING

```
100 DIM A$(1)
110 PRINT "YOU ARE A BRITISH DIPLOMAT BEING HELD CAPTIVE
BY A TERRORIST GROUP."
120 PRINT "SOMEONE JUST SLIPPED YOU A SECRET MESSAGE. "
130 PRINT "DECODE IT AND YOU CAN ESCAPE. HURRY THE
TERRORISTS ARE RETURNING!"
140 PRINT "HINT-CROSS OUT WORDS THAT RHYME"
150 PRINT "THE CODE IS....."
160 PRINT
170 PRINT "PSATRMATEFATSPATSCATLBATERATTYATTHATEZATRDATA"
180 INPUT A$
190 IF A$="A" THEN 220
200 PRINT "◆◆◆BANG BANG THEY SHOT YOU◆◆◆"
210 GOTO 230
220 PRINT "KEY IS UNDER THE BOOK"
230 END
```

Note: Line 190: Notice that a string variable can be used in an IF . . . THEN statement too.

---

# TEN SPEED

---

The computer can help you save toward a goal. This program calculates the time it will take you to save for a ten speed bicycle.

If you change values for the interest rate, the total amount needed, and the monthly amount, you can use the program to calculate how long it will take to save for a vacation, for college, or anything else you want.

## SAMPLE RUN

YOU HAVE DECIDED TO SAVE FOR A NEW TEN SPEED  
IF THE BANK PAYS 5 1/2 % INTEREST COMPOUNDED MONTHLY  
HOW LONG WILL IT TAKE TO SAVE FOR YOUR TEN SPEED?  
THIS PROGRAM WILL GIVE YOU THE ANSWER

HOW MUCH WILL YOUR BIKE COST?183.95  
HOW MUCH CAN YOU SAVE A MONTH?15

FOR YOUR \$ 183.95            TEN SPEED, SAVING \$ 15  
A MONTH,  
YOU CAN HAVE \$ 185.45            IN 12 MONTHS.

## PROGRAM LISTING

```
100 PRINT "YOU HAVE DECIDED TO SAVE FOR A NEW TEN SPEED"  
105 PRINT "IF THE BANK PAYS 5 1/2 % INTEREST COMPOUNDED  
MONTHLY"  
110 PRINT "HOW LONG WILL IT TAKE TO SAVE FOR YOUR TEN  
SPEED?"  
115 PRINT "THIS PROGRAM WILL GIVE YOU THE ANSWER"  
120 PRINT  
125 PRINT "HOW MUCH WILL YOUR BIKE COST";  
130 INPUT A  
135 LET R=.055  
140 PRINT "HOW MUCH CAN YOU SAVE A MONTH";  
145 INPUT M  
150 LET T=0  
155 LET N=0  
160 LET N=N+1  
165 LET T=T+M  
170 LET T=T*(1+R/12)  
175 IF T>A THEN 185
```

```
180 GOTO 160
185 PRINT
190 PRINT "FOR YOUR $ ";A;" TEN SPEED, SAVING $ ";M;"
    A MONTH,"
195 PRINT "YOU CAN HAVE $ ";INT(T*100+.5)/100;" IN
    ";N;" MONTHS."
200 END
```

Note: Line 175: The symbol  $>$  means greater than. This IF . . . THEN statement stops the program as soon as T, the total saved, is greater than A, the amount needed.

### VARIABLE LIST

A—Amount needed for purchase  
R—Current interest rate  
M—Monthly savings  
T—Total earnings  
N—Number of months

---

# TIME DIFFERENCES

---

If you live on the West Coast and you want to call your cousin in Boston, you might want to know what time it is in Boston. This program lets you convert from your time to the time anywhere else in the world.

How would you change the program if you lived in New York and wanted to call a friend in Seattle?

## SAMPLE RUN

```
I CAN TELL YOU WHAT TIME IT IS ANYWHERE IN THE WORLD
JUST INPUT THIS INFORMATION
HIT '0' TO STOP.
```

```
YOUR TIME IN HOURS?8
THE TIME DIFFERENCE IN HOURS?3
THEIR TIME IS 11 0'CLOCK
```

```
YOUR TIME IN HOURS?8
THE TIME DIFFERENCE IN HOURS?-3
THEIR TIME IS 5 0'CLOCK
```

## PROGRAM LISTING

```
100 PRINT "I CAN TELL YOU WHAT TIME IT IS
    ANYWHERE IN THE WORLD"
105 PRINT "JUST INPUT THIS INFORMATION"
110 PRINT "HIT '0' TO STOP."
115 PRINT
120 PRINT "YOUR TIME IN HOURS";
130 INPUT T
132 IF T=0 THEN 175
135 PRINT "THE TIME DIFFERENCE IN HOURS";
140 INPUT D
145 LET N=T+D
150 IF N <= 12 THEN 160
155 N=N-12
160 PRINT "THEIR TIME IS ";N;" 0'CLOCK"
165 PRINT
170 GOTO 120
175 END
```

**Note:** Line 150: The symbols  $\leq$  mean less than or equal to. The program will skip to line 160 if N is equal to 12 or any number less than 12.

### VARIABLE LIST

T—Your time

D—Time difference in hours

N—Their time

---

# INTERIOR DECORATING

---

You want to paint your bedroom walls, and you have some paint already. Now, you are not sure you have enough paint. Use this program to find out if you need to buy any more.

Can you change the program to estimate your needs for more than one room?

## SAMPLE RUN

YOU ARE PAINTING YOUR ROOM AND NEED  
TO FIND OUT IF YOU HAVE ENOUGH  
PAINT. INPUT THE FOLLOWING

LENGTH AND WIDTH OF ROOM?12,20  
SQ FT PER GALLON OF PAINT?100  
GALLONS OF PAINT YOU HAVE NOW?2

BUY MORE PAINT, YOU NEED 3.12 GALLONS MORE

## PROGRAM LISTING

```
100 PRINT "YOU ARE PAINTING YOUR ROOM AND NEED"  
105 PRINT "TO FIND OUT IF YOU HAVE ENOUGH"  
110 PRINT "PAINT. INPUT THE FOLLOWING"  
115 PRINT  
120 PRINT "LENGTH AND WIDTH OF ROOM";  
125 INPUT L,W  
130 PRINT "SQ FT PER GALLON OF PAINT";  
135 INPUT S  
140 PRINT "GALLONS OF PAINT YOU HAVE NOW";  
145 INPUT B  
147 LET H=8  
150 PRINT  
155 LET A=2*L*H+2*W*H  
160 LET G=A/S  
165 IF B <= G THEN 180  
170 PRINT "YOU ARE O.K., YOU ONLY NEED ";G;" GALLONS."  
175 GOTO 185  
180 PRINT "BUY MORE PAINT. YOU NEED ";G-B;" GALLONS  
MORE."  
185 END
```

**Note: Line 155: Calculates the square feet in your room.  
Line 160: Calculates the gallons of paint needed.**

**VARIABLE LIST**

**L—Length**

**W—Width**

**S—Square feet per gallon**

**B—Gallons of paint you have now**

**A—Area of your room**

**G—Gallons needed**

**H—Height of room—set to 8 in this problem**

---

# DRIVER'S TEST

---

When you take the test for your driver's license, you have to estimate the braking distance required to stop a moving car. To practice, try taking our driving test. Remember, it takes a lot farther to stop a moving car than you might think!

## SAMPLE RUN

THIS PROGRAM HELPS YOU DECIDE  
WHEN TO PUT ON THE BRAKES

MI/HR CAR IS TRAVELING?25  
YOUR ESTIMATE OF FEET REQUIRED TO STOP?35  
YOU WOULD HAVE STOPPED TOO SOON  
IT TAKES 20.8765 FEET TO STOP  
AGAIN?Y

MI/HR CAR IS TRAVELING?60  
YOUR ESTIMATE OF FEET REQUIRED TO STOP?100  
YOU WOULD HAVE CRASHED!  
IT TAKES 120.248 FEET TO STOP  
AGAIN?N

## PROGRAM LISTING

```
100 DIM A$(1)
105 PRINT "THIS PROGRAM HELPS YOU DECIDE"
110 PRINT "WHEN TO PUT ON THE BRAKES"
115 PRINT
120 PRINT "MI/HR CAR IS TRAVELING";
125 INPUT S
130 LET S=S*5280/3600
135 D=S ** 2/64.4
140 PRINT "YOUR ESTIMATE OF FEET REQUIRED TO STOP";
145 INPUT F
150 IF F<D-2 THEN 170
155 IF F>D+2 THEN 180
160 PRINT "YOU WERE CLOSE ENOUGH - GOOD DRIVING!"
165 GOTO 190
170 PRINT "YOU WOULD HAVE CRASHED!"
175 GOTO 190
180 PRINT "YOU WOULD HAVE STOPPED TOO SOON"
185 GOTO 190
```

```
190 PRINT "IT TAKES ";D;" FEET TO STOP"  
195 PRINT "AGAIN";  
200 INPUT A$  
205 PRINT  
210 IF A$="Y" THEN 120  
215 END
```

Note: Line 150: Notice that an IF . . . THEN statement can use a numeric expression. The program computes the result of the numeric expression and then applies the IF . . . THEN test.

Lines 150-155: These statements are used together in order to determine if the feet estimated are within 4 feet of the actual feet required to stop.

Lines 195-210: These statements are a clever way of finding out if the person running the program wants to continue. Compare this with the method used in Chapter 4.

#### VARIABLE LIST

S—Miles/Hour converted to Feet/Sec

D—Feet required to stop

F—Input estimate of feet required to stop

---

# SPELLING

---

The computer can help you and your friends solve spelling puzzles. In this game, you have to supply the missing letters needed to complete the word.

Try writing your own program using a different word. You could also add other words to this program.

## SAMPLE RUN

TRY TO FIGURE OUT THE MISSING LETTERS IN THIS WORD

```
C_ROM_?  
THE FIRST LETTER IS?T  
WRONG, TRY AGAIN  
THE FIRST LETTER IS?H  
RIGHT - THE SECOND LETTER IS?R  
WRONG - TRY AGAIN  
THE SECOND LETTER IS?E  
RIGHT, YOU SPELLED CHROME
```

## PROGRAM LISTING

```
100 DIM A$(1),B$(1)  
110 PRINT "TRY TO FIGURE OUT THE MISSING LETTERS IN  
THIS WORD"  
120 PRINT  
130 PRINT "C_ROM_?"  
140 PRINT "THE FIRST LETTER IS";  
150 INPUT A$  
160 IF A$="H" THEN 180  
170 GOTO 240  
180 PRINT "RIGHT - ";  
190 PRINT "THE SECOND LETTER IS";  
200 INPUT B$  
210 IF B$="E" THEN 260  
220 PRINT "WRONG - TRY AGAIN"  
230 GOTO 190  
240 PRINT "WRONG, TRY AGAIN"  
250 GOTO 140  
260 PRINT "RIGHT, YOU SPELLED CHROME"  
270 END
```

**Note:** Line 160: *If you input "H" then the computer goes to line 180; otherwise it drops down to line 170.*

---

# YOU ARE A DETECTIVE

---

Each time you run through this program, you can have a different adventure. Watch out though, only one path is correct.

You can write your own mystery programs too. Just think of a story and use the IF . . . THEN statements to send your detective on different adventures.

## SAMPLE RUN

YOU ARE A DETECTIVE. YOU RECEIVE A CALL FROM MR. DREW. HE ASKS YOU TO COME OVER. HE BELIEVES HIS LIFE IS IN DANGER. IF YOU DECIDE TO GO TO HIS HOUSE TYPE YES. IF YOU DECLINE TYPE NO.

?YES

MR. DREW MEETS YOU AT HIS HOUSE. HE TELLS YOU HE SUSPECTS HIS WIFE IS TRYING TO KILL HIM. HE ASKS YOU TO STAY FOR DINNER SO YOU CAN MEET HER AS WELL AS HIS NIECE LUCY AND A FRIEND DR. QUIMBY. IF YOU ACCEPT HIS INVITATION TYPE MDREW IF YOU DECIDE TO GO HOME TO WATCH YOUR FAVORITE TV SHOW SAYING YOU'LL CALL MR. DREW IN THE MORNING TYPE CALL

?MDREW

MRS. DREW, A RATHER PLUMP PLEASANT WOMAN IS VERY GRACIOUS AT DINNER. AFTER DINNER MR. DREW OFFERS HIS GUESTS A GLASS OF BRANDY. SINCE YOU DECLINE YOU DECIDE TO STROLL AROUND THE GARDEN. SUDDENLY YOU HEAR A CRASH FOLLOWED BY A SCREAM! YOU RUSH IN TO FIND MR. DREW HAS BEEN POISONED! THE POLICE ARE CALLED. BEFORE THEY ARRIVE - YOU DECIDE TO INTERVIEW THE WITNESSES. IF YOU CHOOSE TO SPEAK TO THEM AS A GROUP, TYPE GROUP.

IF YOU INTERVIEW THEM INDIVIDUALLY TYPE ONE.

?ONE

DURING YOUR INTERROGATION OF MR. DREW'S NIECE, SHE MAKES A FATAL SLIP. YOU DISCOVER SHE HAD THE OPPORTUNITY TO PUT ARSENIC IN MR. DREW'S BRANDY BECAUSE SHE WANTED HER INHERITANCE MONEY NOW! WITH

THE POLICE COOPERATION SHE IS  
ARRESTED AND CHARGED WITH HIS  
MURDER. GREAT GOING! MRS. DREW  
REWARDS YOU HANDSOMLY!

### PROGRAM LISTING

```
60 DIM X$(10),Y$(10),Z$(10)
100 PRINT "YOU ARE A DETECTIVE. YOU RECEIVE A CALL"
110 PRINT "FROM MR. DREW. HE ASKS YOU TO COME"
120 PRINT "OVER. HE BELIEVES HIS LIFE IS IN DANGER."
130 PRINT "IF YOU DECIDE TO GO TO HIS HOUSE TYPE YES."
150 PRINT "IF YOU DECLINE TYPE NO."
160 INPUT X$
165 IF X$="YES" THEN 200
170 IF X$="NO" THEN 300
180 PRINT "I NEED A YES OR NO ANSWER. TRY AGAIN."
190 GOTO 160
195 REM♦♦ DECIDE TO MEET MR DREW.
200 PRINT "MR. DREW MEETS YOU AT HIS HOUSE. HE TELLS"
210 PRINT "YOU HE SUSPECTS HIS WIFE IS TRYING TO"
220 PRINT "KILL HIM. HE ASKS YOU TO STAY FOR"
230 PRINT "DINNER SO YOU CAN MEET HER AS WELL"
240 PRINT "AS HIS NIECE LUCY AND A FRIEND DR. QUIMBY."
260 PRINT "IF YOU ACCEPT HIS INVITATION TYPE MDREW"
270 PRINT "IF YOU DECIDE TO GO HOME TO WATCH YOUR"
280 PRINT "FAVORITE TV SHOW SAYING YOU'LL CALL"
290 PRINT "MR. DREW IN THE MORNING TYPE CALL"
292 INPUT Y$
294 IF Y$="MDREW" THEN 400
296 IF Y$="CALL" THEN 500
297 PRINT "I NEED A MDREW OR CALL - TRY AGAIN."
298 GOTO 292
300 PRINT "YOU OBVIOUSLY AREN'T VERY INTERESTED"
310 PRINT "IN KEEPING UP YOUR BUSINESS. SOON"
320 PRINT "YOU ARE FORCED TO FIND ANOTHER JOB."
330 PRINT "BETTER LUCK NEXT TIME!"
340 GOTO 999
400 REM♦♦DECIDE TO STAY FOR DINNER
410 PRINT "MRS. DREW, A RATHER PLUMP PLEASANT WOMAN"
420 PRINT "IS VERY GRACIOUS AT DINNER. AFTER DINNER"
430 PRINT "MR. DREW OFFERS HIS GUESTS A GLASS OF"
440 PRINT "BRANDY. SINCE YOU DECLINE YOU DECIDE"
450 PRINT "TO STROLL AROUND THE GARDEN. SUDDENLY"
460 PRINT "YOU HEAR A CRASH FOLLOWED BY A SCREAM!"
465 PRINT "YOU RUSH IN TO FIND MR. DREW HAS BEEN "
470 PRINT "POISONED! THE POLICE ARE CALLED. BEFORE"
475 PRINT "THEY ARRIVE - YOU DECIDE TO INTERVIEW THE
WITNESSES."
```

```

476 PRINT "IF YOU CHOOSE TO SPEAK TO"
480 PRINT "THEM AS A GROUP, TYPE GROUP."
490 PRINT "IF YOU INTERVIEW THEM INDIVIDUALLY TYPE
ONE."
491 INPUT Z$
492 IF Z$="GROUP" THEN 600
493 IF Z$="ONE" THEN 700
494 PRINT "I NEED A GROUP OR ONE - TRY AGAIN,"
495 GOTO 491
499 REM♦♦DIDN'T STAY FOR DINNER
500 PRINT "YOU CALL THE DREW'S HOUSE IN THE MORNING."
510 PRINT "THE POLICE ANSWER THE PHONE. MR. DREW"
520 PRINT "WAS MURDERED LAST NIGHT. HE WAS POISONED"
530 PRINT "AT DINNER. YOU ARE WANTED AS A WITNESS"
540 PRINT "BUT YOUR DETECTIVE SERVICES ARE NOT
REQUIRED."
550 PRINT "TOO BAD!"
560 GOTO 999
600 REM♦♦INTERVIEW THE GROUP
605 PRINT "WHEN YOU INTERVIEW THE GROUP, YOU FIND"
610 PRINT "OUT NOTHING USEFUL. YOU ARE FORCED"
620 PRINT "TO TURN THE CASE OVER TO THE POLICE"
630 PRINT "WHEN THEY ARRIVE. OUT OF LUCK!"
640 GOTO 999
699 REM♦♦INTERVIEW INDIVIDUALLY
700 PRINT "DURING YOUR INTERROGATION OF MR. DREW'S"
710 PRINT "NIECE, SHE MAKES A FATAL SLIP."
720 PRINT "YOU DISCOVER SHE HAD THE OPPORTUNITY"
730 PRINT "TO PUT ARSENIC IN MR. DREW'S "
740 PRINT "BRANDY BECAUSE SHE WANTED"
750 PRINT "HER INHERITANCE MONEY NOW! WITH"
760 PRINT "THE POLICE COOPERATION SHE IS"
770 PRINT "ARRESTED AND CHARGED WITH HIS"
780 PRINT "MURDER. GREAT GOING! MRS. DREW"
790 PRINT "REWARDS YOU HANDSOMLY!"
999 END

```

Note: Line 165: Notice that you can use words in your IF . . . THEN statements too.

---

# 7 FUNCTIONS (INT, RND, ABS)

## Doing Complex Things the Easy Way

---

Functions are programs built-in to the computer that you can use to make it easy for you to do complicated operations in your own programs.

The functions you need for games are:

**INT** Integer function—Changes any decimal number to a whole number.

$$\text{INT}(6.15) = 6$$

**RND** Random function—Generates random numbers from 0 to 1.

**ABS** Absolute value function—Eliminates the minus sign in a negative number.

$$\text{ABS}(-10) = 10$$

### PROBLEMS

1. Soccer
2. Shell Game
3. Dice Game
4. Number Guessing Game
5. Number Sequences
6. How Warm Is Your Heart?
7. War

---

# SOCCER

---

One of the rules of AYSO (American Youth Soccer Organization) soccer is that everyone on the team must play at least 2 quarters. If all 15 players show up for a game, it is difficult to have everyone play an equal amount of time. This program can help you out. It calculates the number of quarters everyone can play, depending on how many show up for the game.

How can you change the program for your basketball team?

Alter the program to tell you how many innings everyone can play on your Little League team.

## SAMPLE RUN

```
I WILL FIND OUT HOW MANY QUARTERS
EACH MEMBER OF THE TEAM CAN PLAY
NUMBER OF PLAYERS FOR THIS GAME?15
EVERYONE PLAYS 2    QUARTERS
WHILE 14    PLAYERS WILL PLAY 3    QUARTERS
```

## PROGRAM LISTING

```
100 PRINT "I WILL FIND OUT HOW MANY QUARTERS"
105 PRINT "EACH MEMBER OF THE TEAM CAN PLAY"
110 PRINT "NUMBER OF PLAYERS FOR THIS GAME";
115 INPUT N
120 LET P=11*4
125 LET A=P/N
127 IF A >= 4 THEN 155
128 LET A=INT(A)
135 LET Q=P-A*N
140 PRINT "EVERYONE PLAYS ";A;" QUARTERS"
145 PRINT "WHILE ";Q;" PLAYERS WILL PLAY ";A+1;"
    QUARTERS"
150 GOTO 160
155 PRINT "EVERYONE NEEDS TO PLAY 4 QUARTERS"
160 END
```

**Note:** Line 125: The INT function forces the result of P/N to be a whole number, or integer. It cuts off any fractions without rounding.

ex.  $\text{INT}(3.25) = 3$

$\text{INT}(3.95) = 3$

### VARIABLE LIST

N—Number of players at this game

P—Total number of quarters available for the game

A—Number of quarters needed for everyone to play

Q—Extra quarters some players can play

---

# SHELL GAME

---

In this game, you play the original carnival shell game with the computer. Needless to say, this computer's "hand" is definitely quicker than your eye!

Design your own slight of hand game or have your friends guess a prize behind door number 1, 2, or 3.

## SAMPLE RUN

```
READY TO PLAY THE SHELL GAME?
SEE IF YOU CAN GUESS WHICH CUP HAS THE BEAN
OK PLACE YOUR BET?25
WHICH CUP HAS THE BEAN 1 2 OR 3?
?3
YOU ARE RIGHT!
YOU WON $ 50
YOUR TOTAL WINNINGS ARE $ 50
WANT TO TRY AGAIN?Y
OK PLACE YOUR BET?25
WHICH CUP HAS THE BEAN 1 2 OR 3?
?2
YOU ARE RIGHT!
YOU WON $ 50
YOUR TOTAL WINNINGS ARE $ 100
WANT TO TRY AGAIN?Y
OK PLACE YOUR BET?25
WHICH CUP HAS THE BEAN 1 2 OR 3?
?1
I FOOLED YOU THE BEAN WAS IN CUP 2
SORRY . . . YOU LOST $ 25
YOUR TOTAL WINNINGS ARE $ 75
WANT TO TRY AGAIN?N
```

## PROGRAM LISTING

```
100 DIM A$(1)
105 PRINT "READY TO PLAY THE SHELL GAME?"
110 PRINT "SEE IF YOU CAN GUESS WHICH CUP HAS THE BEAN"
115 LET T=0
120 PRINT "OK PLACE YOUR BET";
125 INPUT B
130 LET R=INT(RND(1)*3+1)
135 PRINT "WHICH CUP HAS THE BEAN 1 2 OR 3?"
```

```

140 INPUT A
145 IF A=R THEN 155
150 GOTO 185
155 PRINT "YOU ARE RIGHT!"
160 LET B=B*2
165 LET T=T+B
170 PRINT "YOU WON $ ";B
175 PRINT "YOUR TOTAL WINNINGS ARE $ ";T
180 GOTO 205
185 PRINT "I FOOLED YOU. THE BEAN WAS IN CUP ";R
190 T=T-B
195 PRINT "SORRY...YOU LOST $ ";B
200 PRINT "YOUR TOTAL WINNINGS ARE $ ";T
205 PRINT "WANT TO TRY AGAIN";
210 INPUT AS
215 IF AS="Y" THEN 120
220 END

```

Note: Line 130: This statement gets a random number from 1 to 3. The RND(0) function gives a number from 0 to .9999. Try calculating this expression yourself using different random numbers. This is called playing computer. If you want a number from 1 to 6, just change the 3 to a 6.

### VARIABLE LIST

T—Total winnings  
 B—Bet  
 R—Cup with the bean  
 A—Your guess of the cup

### COMPUTER NOTES

The format for the RND function varies with each computer. See Appendix A for the proper format for your computer.

---

# DICE GAME

---

This program pretends that the computer can throw dice. The person playing the game starts with some money, then wins or loses depending on the throw of the dice as follows:

Dice throw	Original amount of money is:
6 or 12	Tripled
8 or 9	Doubled
7 or 11	Lost
All others	No change

Try changing the betting odds for the game.

## SAMPLE RUN

READY TO PLAY THE DICE GAME?

WHAT IS YOUR BET?10  
DICE THROW IS 5  
NO CHANGE  
YOUR TOTAL WINNINGS ARE 0  
AGAIN?Y

WHAT IS YOUR BET?25  
DICE THROW IS 7  
YOU LOST!  
YOUR TOTAL WINNINGS ARE-25  
AGAIN?Y

WHAT IS YOUR BET?25  
DICE THROW IS 7  
YOU LOST!  
YOUR TOTAL WINNINGS ARE-50  
AGAIN?Y

WHAT IS YOUR BET?50  
DICE THROW IS 8  
YOU WIN! YOU DOUBLED YOUR BET!  
YOUR TOTAL WINNINGS ARE 50  
AGAIN?Y

```
WHAT IS YOUR BET?10
DICE THROW IS 5
NO CHANGE
YOUR TOTAL WINNINGS ARE 50
AGAIN?N
```

### PROGRAM LISTING

```
100 DIM A$(1)
105 LET T=0
110 PRINT "READY TO PLAY THE DICE GAME?"
115 PRINT
120 PRINT "WHAT IS YOUR BET";
125 INPUT M
130 LET A=INT(RND(1)*6+1)
135 LET B=INT(RND(1)*6+1)
140 LET C=A+B
145 PRINT "DICE THROW IS ";C
150 IF C=6 THEN 195
155 IF C=12 THEN 195
160 IF C=8 THEN 210
165 IF C=9 THEN 210
170 IF C=7 THEN 225
175 IF C=11 THEN 225
180 PRINT "NO CHANGE"
185 LET M=0
190 GOTO 240
195 PRINT "YOU WIN! YOU TRIPLED YOUR BET!"
200 LET M=M*3
205 GOTO 240
210 LET M=M*2
215 PRINT "YOU WIN! YOU DOUBLED YOUR BET!"
220 GOTO 240
225 PRINT "YOU LOST!"
230 LET T=T-M
235 GOTO 245
240 LET T=T+M
245 PRINT "YOUR TOTAL WINNINGS ARE ";T
250 PRINT "AGAIN";
255 INPUT A$
260 IF A$="Y" THEN 115
265 END
```

Note: Lines 130, 135: These statements work as if the computer was rolling the dice. They set the numeric variables A and B to a random number between 1 and 6.

## VARIABLE LIST

- M—Your bet
- A—Die throw #1
- B—Die throw #2
- C—Total of both dice
- T—Total winnings

---

# NUMBER GUESSING GAME

---

In this game the computer picks a number from 1 to 100, then 2 players try to guess the number. The player closest to the number wins 10 points.

What if the number was from 1 to 200? What would you change?

## SAMPLE RUN

```
WELCOME TO THE NUMBER GUESSING GAME
THIS IS A GAME FOR TWO PLAYERS
THE COMPUTER PICKS A NUMBER FROM 1 TO 100
EACH PLAYER GUESSES A NUMBER
THE PLAYER CLOSEST TO THE CORRECT NUMBER GETS 10 POINTS
A PLAYER GETS 50 POINTS IF THE PLAYER GUESSES THE NUMBER
```

```
OK INPUT YOUR GUESS
PLAYER 1?30
PLAYER 2?70
PLAYER 2 WINS
THE COMPUTER'S NUMBER WAS 64
THE SCORE IS - PLAYER 1 0
                PLAYER 2 10
```

```
AGAIN?Y
OK INPUT YOUR GUESS
PLAYER 1?750
PLAYER 2?100
PLAYER 1 WINS
THE COMPUTER'S NUMBER WAS 51
THE SCORE IS - PLAYER 1 10
                PLAYER 2 10
```

```
AGAIN?Y
OK INPUT YOUR GUESS
PLAYER 1?20
PLAYER 2?80
PLAYER 1 WINS
THE COMPUTER'S NUMBER WAS 42
THE SCORE IS - PLAYER 1 20
                PLAYER 2 10
```

```
AGAIN?N
```

## PROGRAM LISTING

```
100 DIM A$(1)
105 PRINT "WELCOME TO THE NUMBER GUESSING GAME"
110 PRINT "THIS IS A GAME FOR TWO PLAYERS"
115 PRINT "THE COMPUTER PICKS A NUMBER FROM 1 TO 100"
120 PRINT "EACH PLAYER GUESSES A NUMBER"
125 LET S=0
130 LET T=0
135 PRINT "THE PLAYER CLOSEST TO THE CORRECT NUMBER
GETS 10 POINTS"
140 PRINT "A PLAYER GETS 50 POINTS IF THE PLAYER
GUESSES THE NUMBER"
145 PRINT
150 PRINT "OK INPUT YOUR GUESS"
155 PRINT "PLAYER 1";
160 INPUT P
165 PRINT "PLAYER 2";
170 INPUT N
175 A=INT(RND(1)*100+1)
180 LET B=ABS(P-A)
185 LET C=ABS(N-A)
190 IF B>C THEN 235
195 IF B<C THEN 220
200 PRINT "YOU BOTH WIN"
205 LET S=S+10
210 LET T=T+10
215 GOTO 245
220 PRINT "PLAYER 1 WINS"
225 LET S=S+10
230 GOTO 245
235 PRINT "PLAYER 2 WINS"
240 LET T=T+10
245 PRINT "THE COMPUTER'S NUMBER WAS ";A
250 PRINT "THE SCORE IS - PLAYER 1 ";S
255 PRINT " - PLAYER 2 ";T
260 PRINT "AGAIN";
265 INPUT A$
270 IF A$="Y" THEN 150
275 END
```

Note: Line 175: The computer gets a random number from 1 to 100.

Lines 180, 185: The ABS function eliminates the minus sign. For example, if  $P=5$  and  $A=10$  then  $P-A=-5$ , but the ABS function would change  $-5$  to  $5$ .

Lines 190, 195: The computer calculates which guess is closest to the computer's number.

### VARIABLE LIST

P—Player #1's guess

N—Player #2's guess

A—The computer's number

B—The difference between Player #1 and the computer

C—The difference between Player #2 and the computer

S—Player #1's winnings

T—Player #2's winnings

---

# NUMBER SEQUENCES

---

This program prints a sequence of numbers. See if you can figure out what the next number should be.

Can you fix the program so it keeps track of the number of correct answers?

## SAMPLE RUN

THIS PROGRAM PRINTS A SEQUENCE OF NUMBERS  
SEE IF YOU CAN FIGURE OUT WHAT THE NEXT NUMBER SHOULD BE

3 5 7 9 11 ?13  
YOU WERE RIGHT, CONGRATULATIONS!

3 9 27 81 243 ?729  
YOU WERE RIGHT, CONGRATULATIONS!

5 10 20 40 80 ?160  
YOU WERE RIGHT, CONGRATULATIONS!

3 7 11 15 19 ?12  
SORRY, THE NEXT NUMBER IS 23

1 3 9 27 81 ?243  
YOU WERE RIGHT, CONGRATULATIONS!

3 10 17 24 31 ?38  
YOU WERE RIGHT, CONGRATULATIONS!

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM PRINTS A SEQUENCE OF NUMBERS"  
105 PRINT "SEE IF YOU CAN FIGURE OUT WHAT THE NEXT  
NUMBER SHOULD BE"  
110 S=INT(RND(1)*2+1)  
115 A=INT(RND(1)*5+1)  
120 B=INT(RND(1)*10+1)  
125 C=INT(RND(1)*3+1)  
130 IF C=1 THEN 125  
135 I=0  
140 IF S=1 THEN 165  
145 REM** THIS IS A GEOMETRIC SEQUENCE  
150 N=A*C ** I  
155 GOTO 170  
160 REM** THIS IS AN ARITHMETRIC SEQUENCE  
165 N=A+I*B  
170 I=I+1  
175 IF I>5 THEN 190
```

```

177 LET N=INT(N)
180 PRINT N;
185 GOTO 140
190 INPUT M
195 IF M=N THEN 210
200 PRINT "SORRY, THE NEXT NUMBER IS ";N
205 GOTO 110
210 PRINT "YOU WERE RIGHT, CONGRATULATIONS!"
215 GOTO 110
220 END

```

Note: Line 150: The computer calculates a geometric sequence.  
 Line 165: The computer calculates an arithmetic sequence.

### VARIABLE LIST

S—Determines the type of sequence  
 S = 1 arithmetic sequence  
 S = 2 geometric sequence  
 A—For arithmetic sequence—first number  
 For geometric sequence—multiplier  
 B—For arithmetic sequence—common difference  
 C—For geometric sequence—common ratio  
 M—Next number in sequence

---

# HOW WARM IS YOUR HEART?

---

How warm is your heart? Use this program to find out. You type in a number from 1 to 5 and the computer decides how warm you really are. Don't be surprised if the computer changes its mind. After all, computers don't know everything.

You can add responses of your own to this program.

Try writing a program of your own that evaluates someone's personality or athletic ability.

## SAMPLE RUN

```
TO SEE HOW WARM YOUR HEART IS
TYPE A NUMBER FROM 1 TO 5
?1
YOUR HEART IS ON FIRE!!
TYPE A NUMBER FROM 1 TO 5
?5
YOUR HEART IS IN THE NORTH POLE!
TYPE A NUMBER FROM 1 TO 5
?3
YOUR HEART IS IN THE NORTH POLE!
```

## PROGRAM LISTING

```
100 PRINT "TO SEE HOW WARM YOUR HEART IS"
105 PRINT "TYPE A NUMBER FROM 1 TO 5"
110 INPUT N
115 IF N>5 THEN 105
120 S=INT(RND(1)*2)*(-2)+1
125 R=INT(RND(1)*5+1)
130 T=N+S*R
135 IF T>1 THEN 150
140 PRINT "YOUR HEART IS IN THE NORTH POLE!"
145 GOTO 105
150 IF T>2 THEN 165
155 PRINT "YOUR HEART IS IN A DEEP FREEZE!"
160 GOTO 105
165 IF T>3 THEN 180
170 PRINT "YOUR HEART IS THAWING OUT!"
175 GOTO 105
180 IF T>4 THEN 195
185 PRINT "YOUR HEART IS STARTING TO GLOW!"
```

```
190 GOTO 105
195 PRINT "YOUR HEART IS ON FIRE!!"
200 GOTO 105
205 END
```

Note: Line 120: Pretend that you are the computer, and try to calculate what value S can have when this statement is run.

### VARIABLE LIST

- N—Number input
- R—Random number 1 to 5
- S—Determines if random number is added to or subtracted from the number input
- T—Determines which response is printed

---

# WAR

---

Play the card game WAR with the computer. You and the computer each start with 26 cards. When it is your turn, input a number from 1 to 5. If you match with the computer, you have a BATTLE. Input a 1 or 2 to decide the outcome. If you pick the same number as the computer, you win. If it is different, the computer wins. The winner of the BATTLE gets all of the opponent's cards since the last BATTLE. The WAR is won when one of you gets more than 42 cards. Good Luck!

Can you write a program that plays one of your favorite card games?

## SAMPLE RUN

```
WELCOME TO THE GAME WAR
WE EACH START WITH 26 CARDS
NUMBERED 1-5
WHEN IT IS YOUR TURN - INPUT A NUMBER 1 - 5
```

```
YOUR CARD?1
COMPUTERS CARD 1
THIS IS WAR. PICK A 1 OR 2?1
TOSS WAS A 1
YOU WIN 1 CARDS
```

```
YOUR CARD?2
COMPUTERS CARD 4
YOUR CARD?3
COMPUTERS CARD 4
YOUR CARD?2
COMPUTERS CARD 4
YOUR CARD?4
COMPUTERS CARD 1
YOUR CARD?1
COMPUTERS CARD 4
YOUR CARD?4
COMPUTERS CARD 2
YOUR CARD?3
COMPUTERS CARD 4
YOUR CARD?2
COMPUTERS CARD 2
THIS IS WAR. PICK A 1 OR 2?1
TOSS WAS A 1
YOU WIN 8 CARDS
```

## PROGRAM LISTING

```
100 DIM A$(1)
105 PRINT "WELCOME TO THE GAME WAR"
110 PRINT "WE EACH START WITH 26 CARDS"
115 PRINT "NUMBERED 1-5"
120 PRINT "WHEN IT IS YOUR TURN - INPUT A NUMBER 1 - 5"
125 REM♦♦COMPUTERS CARDS
130 LET T=26
135 REM♦♦YOUR CARDS
140 LET S=26
145 LET U=0
150 LET V=0
155 PRINT
160 PRINT "YOUR CARD";
165 INPUT N
170 LET V=V+1
175 C=INT(RND(1)♦5+1)
180 PRINT "COMPUTER'S CARD ";C
185 LET U=U+1
190 IF C=N THEN 230
195 IF U<T THEN 210
200 REM♦♦ USED ALL CARDS IN PILE - TURN PILE OVER
205 LET U=0
210 IF V<S THEN 160
215 LET V=0
220 GOTO 160
225 REM♦♦HAD A MATCH♦♦
230 PRINT "THIS IS WAR. PICK A 1 OR 2";
235 INPUT P
240 R=INT(RND(1)♦2+1)
245 PRINT "TOSS WAS A ";R
250 IF R=P THEN 280
255 PRINT "I WIN ";V;" CARDS"
260 LET S=S-V
265 LET T=T+V
270 IF T>42 THEN 325
275 GOTO 145
280 PRINT "YOU WIN ";U;" CARDS"
285 LET S=S+U
290 LET T=T-U
295 IF S>42 THEN 310
300 GOTO 145
305 REM♦♦PLAYER WINS GAME♦♦
310 PRINT "YOU WIN GAME"
315 GOTO 330
320 REM♦♦COMPUTER WINS♦♦
325 PRINT "I WIN GAME"
330 PRINT "I HAVE ";T;" CARDS"
335 PRINT "YOU HAVE ";S;" CARDS"
```

```
340 PRINT
345 PRINT "TRY AGAIN(Y/N)";
350 INPUT AS
355 IF AS="Y" THEN 105
360 PRINT
365 PRINT "BETTER LUCK NEXT TIME!"
370 END
```

Note: Line 175: This statement randomly assigns a number from 1 to 5 to the numeric variable C. This is how a programmer makes the computer play games.

Lines 125, 135: Notice how the REM statement is used to identify variables. This makes it easier to change or fix a program later.

### VARIABLE LIST

- T—Number of cards the computer has
- S—Number of cards the player has
- U—Number of cards computer has turned up
- V—Number of cards the player has turned up
- C—Computer's number (1-5)
- N—Player's number (1-5)
- P—Player's guess during a battle (1-2)
- R—Random number determining winner of a battle

# 8

---

## FOR . . . NEXT

### The Computer Repeats Itself

---

The FOR . . . NEXT statements allow you to tell the computer to repeat some lines in your program as many times as you want. In every FOR . . . NEXT loop, the FOR statement is the beginning point of the loop and the NEXT statement is always the last statement in the loop.

#### PROBLEMS

1. Fancy Counting
2. Blast Off
3. Pinball
4. Paper, Rock, Scissors
5. Ring the Bell

---

# FANCY COUNTING

---

This program counts by fives. It is very similar to one in Chapter 3, but this one uses a new programming technique. Which program do you like better?

Can you change this program to count by 2's or by 10's?

## SAMPLE RUN

THIS PROGRAM COUNTS BY FIVES - WATCH

5  
10  
15  
20  
25  
30  
35  
40  
45  
50

HOW DID YOU LIKE THAT?

## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM COUNTS BY FIVES - WATCH"  
110 FOR I=5 TO 50 STEP 5  
120 PRINT I  
130 NEXT I  
140 PRINT "HOW DID YOU LIKE THAT?"  
150 END
```

Note: Lines 110-130: This is called a FOR . . . NEXT loop. The computer executes lines 110, 120, and 130 and then goes back to 110 and starts over. Each time line 110 is executed, the variable I is increased by 5. When I is greater than 50 the computer skips around the loop and goes to line 140.

---

# BLAST OFF

---

This program counts backward from 10 to 1 and then Blasts Off!  
You can change the count-down sequence by changing line 30.

## SAMPLE RUN

```
HELLO, I AM READY TO BLAST OFF  
HERE IS THE COUNT DOWN
```

```
10  
9  
8  
7  
6  
5  
4  
3  
2  
1  
BLAST OFF
```

## PROGRAM LISTING

```
10 PRINT "HELLO, I AM READY TO BLAST OFF"  
20 PRINT "HERE IS THE COUNT DOWN"  
30 FOR I=10 TO 1 STEP -1  
40 PRINT I  
50 NEXT I  
60 PRINT "BLAST OFF"  
70 END
```

Note: Line 30: The FOR . . . NEXT statement can also use a minus step like this one. Then the computer stops the loop when  $I = 1$ .

---

# PINBALL

---

The computer can play Pinball too. The highest possible score in this game is 50; see how well you do. This game makes some noise if your terminal has a bell.

Try changing the characters (the letters or numbers) displayed during the game. You might use names or nouns instead of numbers.

Try changing the program so the player has to pay a quarter to play, and then keep track of the total amount of money the player spends.

## SAMPLE RUN

TYPE A NUMBER TO START YOUR BALL OFF?4

◆◆TWO◆◆

◆◆FIVE◆◆

◆◆SIX◆◆

◆◆TWO◆◆

◆◆FIVE◆◆

◆◆THREE◆◆

◆◆SIX◆◆

◆◆FOUR◆◆

◆◆SIX◆◆

◆◆TWO◆◆

YOUR SCORE IS 41

YOUR TOTAL SCORE IS 41

## PROGRAM LISTING

```
100 DIM A$(1)
105 T=0
110 PRINT "TYPE A NUMBER TO START YOUR BALL OFF";
115 INPUT A
120 S=0
125 FOR I=1 TO 10
130 R=INT(RND(1)*6+1)
135 S=S+R
140 IF R>1 THEN 155
145 PRINT TAB(R*10),CHR$(7),"◆◆ONE◆◆"
150 GOTO 220
155 IF R>2 THEN 170
160 PRINT TAB(R*10),CHR$(7),"◆◆TWO◆◆"
```

```

165 GOTO 220
170 IF R>3 THEN 185
175 PRINT TAB(R*10),CHR$(7),"◆◆THREE◆◆"
180 GOTO 220
185 IF R>4 THEN 200
190 PRINT TAB(R*10),CHR$(7),"◆◆FOUR◆◆"
195 GOTO 220
200 IF R>5 THEN 215
205 PRINT TAB(R*10),CHR$(7),"◆◆FIVE◆◆"
210 GOTO 220
215 PRINT TAB(R*10),CHR$(7),"◆◆SIX◆◆"
220 NEXT I
225 PRINT "YOUR SCORE IS",S
230 T=T+S
235 PRINT "YOUR TOTAL SCORE IS",T
240 PRINT "AGAIN";
245 INPUT A$
250 IF A$="Y" THEN 110
255 END

```

Note: Line 145: CHR\$(7) rings a bell on your terminal.

Line 150: You can use a GOTO statement to jump to the end of the loop.

Line 125, 220: Notice how many lines of code are included in this FOR . . . NEXT loop.

### VARIABLE LIST

T—Total score

S—Game score

R—Points for each round

---

# PAPER, ROCK, SCISSORS

---

You can play this game with your computer. Remember . . . rock breaks scissors, scissors cut paper and paper covers rock.

Try changing the program so the computer keeps score.

## SAMPLE RUN

```
PLAY THE GAME - PAPER, ROCK, SCISSORS - WITH THE COMPUTER
MAKE YOUR CHOICE AND THEN TYPE IN
R FOR ROCK, S FOR SCISSORS, P FOR PAPER
TYPE X TO STOP
?R
YOU WIN ** COMPUTER HAD S
?S
COMPUTER WON ** COMPUTER HAD R
?S
WE MATCHED
?R
YOU WIN ** COMPUTER HAD S
?S
YOU WIN ** COMPUTER HAD P
?P
YOU WIN ** COMPUTER HAD R
?X
```

## PROGRAM LISTING

```
100 DIM B$(4),A$(1)
105 LET B$ = "PSRX"
110 PRINT "PLAY THE GAME - PAPER, ROCK, SCISSORS - WITH
THE COMPUTER"
115 PRINT "MAKE YOUR CHOICE AND THEN TYPE IN"
120 PRINT "R FOR ROCK, S FOR SCISSORS, P FOR PAPER"
125 PRINT "TYPE X TO STOP"
130 LET C = INT ( RND (1) * 3 + 1)
135 INPUT A$
140 REM **FIND OUT WHAT LETTER WAS TYPED IN
145 FOR I = 1 TO 4
150 IF A$ = MID$(B$,I,1) THEN 170
155 NEXT I
160 PRINT "I NEED A R,S,P,X PLEASE"
165 GOTO 135
170 IF I = 4 THEN 250
```

```

175 REM **GOT A MATCH
180 IF C = I THEN 240
185 REM **GOT P,R OR R,P
190 IF C + I = 4 THEN 225
195 IF C > I THEN 210
200 PRINT "YOU WIN ** COMPUTER HAD "; MID$(B$,C,1)
205 GOTO 130
210 PRINT "COMPUTER WON ** COMPUTER HAD "; MID$(B$,C,1)
215 GOTO 130
220 REM **PLAY WON
225 IF C > I THEN 200
230 REM **COMPUTER WON
235 GOTO 210
240 PRINT "WE MATCHED"
245 GOTO 130
250 END

```

Note: Lines 145-155: These statements translate the input character into a number which the program can use.

Lines 160-165: This "error routine" is needed in case the input character is wrong.

### VARIABLE LIST

C—Computer's choice  
I—Player's choice in numeric  
A\$—Character input by player

### COMPUTER NOTES

HP-2000 BASIC      Replace line 150 with:  
                         150 IF A\$=B\$(I,I) THEN 170  
                         Replace line 200 with:  
                         200 PRINT "YOU WIN \*\* COMPUTER  
                         HAD "; B\$(C,C)  
                         Replace line 210 with:  
                         210 PRINT "COMPUTER WON \*\* COM-  
                         PUTER HAD "; B\$(C,C)

## Atari-BASIC

Replace line 150 with:

```
150 IF A$ = B$(I,I) THEN 170
```

Replace line 200 with:

```
200 PRINT "YOU WIN ** COMPUTER  
HAD ", B$(C,C)
```

Replace line 210 with:

```
210 PRINT "COMPUTER WON ** COM-  
PUTER HAD "; B$(C,C)
```

## TI-BASIC

Replace line 150 with:

```
150 IF A$ = SEG$(B$,I,1) THEN 170
```

Replace line 200 with:

```
200 PRINT "YOU WIN ** COMPUTER  
HAD "; SEG$(B$,C,1)
```

Replace line 210 with:

```
210 PRINT "COMPUTER WON ** COM-  
PUTER HAD "; SEG$(B$,C,1)
```

---

# RING THE BELL

---

Now is your chance to see how strong you are. In this game you are trying to ring the bell by hitting the platform with a sledgehammer—all pretend of course! The computer tells you how well you do.

Try changing the game by adding carnival prizes after a certain number of points.

## SAMPLE RUN

YOU ARE TRYING TO WIN THE STRONG MAN CONTEST AT THE  
LOCAL CARNIVAL  
YOU ARE GOING TO SWING THE SLEDGE HAMMER AND TRY TO  
RING THE BELL  
TO SWING, TYPE A LETTER FROM A-Z  
THIS WILL COST YOU .25 - PUT YOUR QUARTER ON THE TERMINAL

NOW MAKE YOUR SWING?E  
YOU CAN HARDLY PICK UP THE HAMMER, YOU SCORED 2  
YOUR TOTAL SCORE IS 2  
YOU HAVE SPENT \$.25  
THIS WILL COST YOU .25 - PUT YOUR QUARTER ON THE TERMINAL

NOW MAKE YOUR SWING?A  
YOUR 90 YEAR OLD AUNT COULD HAVE SCORED 3  
YOUR TOTAL SCORE IS 5  
YOU HAVE SPENT \$.5  
THIS WILL COST YOU .25 - PUT YOUR QUARTER ON THE TERMINAL

NOW MAKE YOUR SWING?P  
YOU MOVED THE WEIGHT UP TO 4  
YOUR TOTAL SCORE IS 9  
YOU HAVE SPENT \$.75

## PROGRAM LISTING

```
100 DIM A$(1),B$(8)
105 LET B$ = "BCPQUVJK"
110 PRINT "YOU ARE TRYING TO WIN THE STRONG MAN CONTEST
      AT THE LOCAL CARNIVAL"
115 PRINT "YOU ARE GOING TO SWING THE SLEDGE HAMMER AND
      TRY TO RING THE BELL"
```

```

120 PRINT "TO SWING, TYPE A LETTER FROM A-Z"
125 LET T = 0
130 LET B = 0
135 PRINT "THIS WILL COST YOU .25 - PUT YOUR QUARTER
ON THE TERMINAL"
136 PRINT
137 PRINT "NOW MAKE YOUR SWING";
140 INPUT A$
145 LET B = B + .25
150 REM **CHECK FOR INPUT EQUAL TO ONE OF CHARACTERS
IN B$
155 FOR I = 1 TO 8
160 IF A$ = MID$(B$,I,1) THEN 175
165 NEXT I
170 GOTO 185
175 LET C = INT ((I + 1) / 2)
180 GOTO 190
185 LET C = 0
190 LET D = INT ( RND (1) * 6 + 1)
195 LET S = C + D
200 ON S GOTO 210,220,250,230,240,260,270
205 GOTO 280
210 PRINT "BETTER EAT YOUR WHEATIES, YOU ONLY SCORED 1"
215 GOTO 300
220 PRINT "YOU CAN HARDLY PICK UP THE HAMMER, YOU
SCORED 2"
225 GOTO 300
230 PRINT "YOU MOVED THE WEIGHT UP TO 4"
235 GOTO 300
240 PRINT "YOU SCORED A 5 - YOU MUST BE PRACTICING
245 GOTO 300
250 PRINT "YOUR 90 YEAR OLD AUNT COULD HAVE SCORED 3"
255 GOTO 300
260 PRINT "YOU MUST BE DOING YOUR PUSHUPS, YOU SCORED 6"
265 GOTO 300
270 PRINT "YOU ALMOST MADE IT - YOU SCORED A 7"
275 GOTO 300
280 PRINT CHR$(7)
285 PRINT "CONGRATULATIONS!! YOU RANG THE BELL!!"
290 PRINT "YOU GET 8 POINTS!!"
300 LET T = T + S
305 PRINT "YOUR TOTAL SCORE IS ",T
310 PRINT "YOU HAVE SPENT $";B
325 GOTO 135
330 END

```

**Note:** Lines 155-180: If the letter entered by the player is one of the 8 letters stored in B\$, this routine adds a bonus to the player's score.

Line 200: This GOTO statement replaces 7 single GOTO's. It is very useful when a variable such as S can have many different values.

### VARIABLE LIST

T—Player's total score

B—Amount player has spent

A\$—Input character

B\$—Contains a list of letters that add a bonus to the player's score

D—Random number (1-6)

C—Bonus point if player enters one of the letters in B\$

S—Player's score for this game

### COMPUTER NOTES

**HP-2000 BASIC**      The multiple GOTO on line 200 has a different format on the HP-200. Replace line 200 with:

```
200 GOTO S OF 210, 220, 250, 230,  
240, 260, 270
```

Replace line 160 with:

```
160 IF A$ = B$(I, I) THEN 175
```

**Atari BASIC**

Replace line 160 with:

```
160 IF A$ = B$(I,I) THEN 175
```

**TI-BASIC**

Replace line 160 with:

```
160 IF A$ = SEG$(B$,I,1) THEN 175
```

---

# 9 FANCY PRINTING (TAB)

## Writing It Your Way

---

This function makes it easy for you to draw designs with the computer. It works like the tab key on a typewriter. It tells the computer to move a certain number of spaces across the line and then print. You can use numbers, variables or numeric expressions inside the ( ).

### PROBLEMS

1. Rectangle
2. Design-a-Scarf
3. HIHO
4. Lifeguard
5. Shape
6. Party
7. Fleet
8. Award



Line 180: This TAB function tells the computer to print the next \* in column (W-1).

### VARIABLE LIST

H—Height of rectangle

W—Width of rectangle

### COMPUTER NOTES

Atari BASIC Does not have the TAB function. Appendix A describes a method of converting these programs for the Atari.



```

160 FOR I = 1 TO H
170 PRINT "-";
180 NEXT I
190 PRINT
200 FOR I = 1 TO H - 2 STEP 2
210 PRINT "I";
220 LET R = INT ( RND (1) * (W - 6)) + 1
230 PRINT TAB( R);I$; TAB( W - 1);"I"
240 PRINT "I"; TAB( W - 1);"I"
250 NEXT I
260 FOR I = 1 TO H
270 PRINT "-";
280 NEXT I
290 END

```

### VARIABLE LIST

I—Characters to use in design  
H—Height of scarf outline  
W—Width of scarf outline  
R—Column to print design

---

# HIHO

---

The TAB function helps make interesting patterns like this one.  
Can you write a program that prints WIN in the shape of a V?

## SAMPLE RUN

WATCH WHAT I CAN MAKE

```
HI           HO
 HI          HO
  HI         HO
   HI        HO
    HI       HO
     HI      HO
      HI     HO
       HI    HO
        HI   HO
         HI  HO
          HI HO
           HI
          HOHI
         HO  HI
        HO   HI
       HO    HI
      HO     HI
     HO      HI
    HO       HI
   HO        HI
  HO         HI
 HO          HI
HO           HI
```

## PROGRAM LISTING

```
100 PRINT "WATCH WHAT I CAN MAKE"
110 PRINT
120 PRINT
130 FOR K=1 TO 9
140 PRINT TAB(K); "HI"; TAB(20-K); "HO"
150 NEXT K
160 PRINT TAB(10); "HI"
170 FOR K=11 TO 20
180 PRINT TAB(20-K); "HO"; TAB(K); "HI"
190 NEXT K
200 END
```

---

# LIFEGUARD

---

You can have some fun with the TAB function too. In this game, you hear a swimmer calling for help and you try to throw him a life ring. If you hit him, you can save him—otherwise—it is Davy Jones's locker for him.

Change the program so it only gives you four tries to save the swimmer.

## SAMPLE RUN

```
YOU MISSED, TRY AGAIN!  
WHERE YOU WANT IT TO LAND?5  
  0  HELP
```

```
YOU MISSED, TRY AGAIN!  
WHERE YOU WANT IT TO LAND?6  
HELP 0
```

```
YOU MISSED, TRY AGAIN!  
WHERE YOU WANT IT TO LAND?3  
  0  HELP
```

```
YOU MISSED, TRY AGAIN!  
WHERE YOU WANT IT TO LAND?5  
THANKS! YOU SAVED MY LIFE!
```

## PROGRAM LISTING

```
100 PRINT "USE THIS PROGRAM TO HELP A DROWNING SWIMMER"  
110 PRINT "TO THROW THE LIFE RING, INPUT 1-10"  
120 PRINT "WHERE YOU WANT IT TO LAND";  
130 LET X=INT(RND(1)*10+1)  
140 INPUT S  
150 IF S=X THEN 230  
160 IF S>X THEN 190  
170 PRINT TAB(S);"0";TAB(X);"HELP"  
180 GOTO 200  
190 PRINT TAB(X);"HELP";TAB(S);"0"  
200 PRINT  
210 PRINT "YOU MISSED, TRY AGAIN!"  
220 GOTO 120  
230 PRINT "THANKS! YOU SAVED MY LIFE!"  
240 END
```

## VARIABLE LIST

S—Column where life ring is thrown

X—Column where swimmer is located

---

# SHAPE

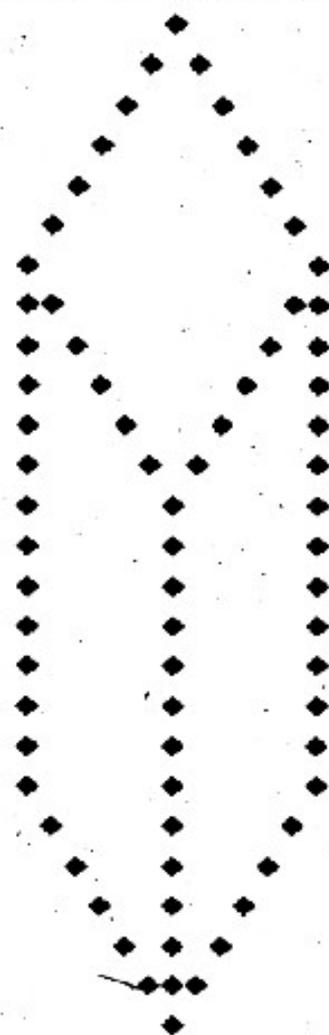
---

You can make 3-dimensional shapes like this one. The secret for success is to be very careful to count out each space.

Once you understand how this program works, modify it to make a shape of your own.

## SAMPLE RUN

THIS PROGRAM PRINTS A VERY FAMILIAR SHAPE



## PROGRAM LISTING

```
100 PRINT "THIS PROGRAM PRINTS A VERY FAMILIAR SHAPE"  
110 PRINT TAB(15); "*"   
120 FOR S=1 TO 6  
130 PRINT TAB(15-S); "*" ; TAB(15+S); "*"   
140 NEXT S
```

```

150 FOR T=5 TO 1 STEP -1
160 PRINT TAB(9);"♦";TAB(15-T);"♦";TAB(15+T);"♦";
    TAB(21);"♦"
170 NEXT T
180 FOR U=1 TO 8
190 PRINT TAB(9);"♦";TAB(15);"♦";TAB(21);"♦"
200 NEXT U
210 FOR V=5 TO 1 STEP -1
220 PRINT TAB(15-V);"♦";TAB(15);"♦";TAB(15+V);"♦"
230 NEXT V
240 PRINT TAB(15);"♦"
250 END

```

**Note:** Notice that each FOR . . . NEXT loop prints a different section of the cube.

---

# PARTY

---

Use this program to print party invitations for your next party. This one asks for your name, date and time of the party and your address.

Try making an invitation of your own.

## SAMPLE RUN

MAKE UP YOUR OWN PARTY INVITATIONS  
YOUR NAME?BECKY  
YOUR ADDRESS?4525 ELMWOOD DRIVE  
DAY OF THE PARTY?MAY 30  
TIME OF THE PARTY?1:00 P.M.  
O.K. HERE IS YOUR INVITATION

PLEASE COME

TO MY

BIRTHDAY PARTY

DATE: MAY 30  
TIME: 1:00 P.M.  
PLACE: 4525 ELMWOOD DRIVE

BY BECKY

## PROGRAM LISTING

```
100 DIM A$(25), B$(25), C$(25), D$(25)
110 PRINT "MAKE UP YOUR OWN PARTY INVITATIONS"
120 PRINT "YOUR NAME":
130 INPUT D$
140 PRINT "YOUR ADDRESS":
150 INPUT A$
```

```

160 PRINT "DAY OF THE PARTY";
170 INPUT BS
180 PRINT "TIME OF THE PARTY";
190 INPUT CS
200 PRINT "O.K. HERE IS YOUR INVITATION"
210 PRINT
211 PRINT
212 PRINT
220 PRINT TAB(15); "PLEASE COME"
221 PRINT
222 PRINT
230 PRINT TAB(18); "TO MY"
231 PRINT
232 PRINT
240 PRINT TAB(14); "BIRTHDAY PARTY"
241 PRINT
242 PRINT
243 PRINT
250 PRINT TAB(15); "DATE: "; BS
260 PRINT TAB(15); "TIME: "; CS
270 PRINT TAB(15); "PLACE: "; AS
280 PRINT
281 PRINT
282 PRINT
290 PRINT TAB(15); "BY "; DS
300 END

```

---

# FLEET

---

Watch this fleet of rocket ships blast off from your video terminal.  
Try adding your own ships to this fleet.

## SAMPLE RUN

WATCH THIS FLEET OF ROCKET SHIPS BLAST OFF!

```
      ^
     I I
     I I
     I I
  /  \
  /  \
```

```
      ^
     I I
     I I
     I I
  /  \
  /  \
```

## PROGRAM LISTING

```
100 PRINT "WATCH THIS FLEET OF ROCKET SHIPS BLAST OFF!"
105 PRINT
106 PRINT
107 PRINT
108 PRINT
109 PRINT
110 LET R=INT(RND(1)*20+2)
120 PRINT TAB(R);"^";TAB(R+15);"^"
125 FOR I=1 TO 3
130 PRINT TAB(R-1);"I I";TAB(R+14);"I I"
135 NEXT I
140 PRINT TAB(R-2);"/_/_/_\";TAB(R+13);"/_/_/_\"
150 FOR J=1 TO 15
160 FOR I=1 TO 100
170 NEXT I
180 PRINT
190 NEXT J
200 GOTO 110
210 END
```

**Note:** Lines 150-190: These statements make the fleet of ships blast off your video screen.

Lines 160-170: Slows the computer down so you can see what happens.





---

# 10 SUBSCRIPTS

## Where to Store Your Information

---

You can store more than one number in a numeric variable by using subscripts. A subscripted variable is written  $A(3)$ , where  $A$  is the variable name and the number inside the  $( )$  is the subscript. The subscript determines which of the numbers stored in the variables to use. For example: if you have 3 numbers in  $A$ , then  $A(1)$  is the first number,  $A(2)$  is the second, etc. You can input, print, and do calculations with subscripted variables just as with other numeric variables.

### PROBLEMS

1. Most Valuable Player
2. Bank Statement
3. Pascal's Triangle
4. Sea Battle

---

# MOST VALUABLE PLAYER

---

Help your coach pick the most valuable player on your soccer team. You rate each of the best 6 players in 3 categories, and this program calculates a MVP score.

Can you fix this program so it can use more than 6 players?

Can you change the program to add another rating category?

## SAMPLE RUN

THIS PROGRAM HELPS YOU FIND THE  
MOST VALUABLE PLAYER ON YOUR SOCCER TEAM  
RATE EACH OF THE 6 PLAYERS YOU THINK ARE THE BEST  
ON THE FOLLOWING CATEGORIES 1 - 5 (5 BEST)

SOCCER SKILLS

ATTITUDE

ATHLETIC ABILITY

THE COMPUTER WILL CALCULATE A MVP SCORE

PLAYER 1?1,2,3

PLAYER 2?2,2,5

PLAYER 3?1,5,5

PLAYER 4?4,3,4

PLAYER 5?2,3,4

PLAYER 6?5,3,4

THE RATINGS ARE AS FOLLOWS

PLAYER	SKILL	ATTITUDE	ABILITY	MVP
1	1	2	3	11
2	2	2	5	18
3	1	5	5	18
4	4	3	4	23
5	2	3	4	17
6	5	3	4	26

## PROGRAM LISTING

```
95 DIM S(6),A(6),G(6),T(6)
100 PRINT "THIS PROGRAM HELPS YOU FIND THE"
110 PRINT "MOST VALUABLE PLAYER ON YOUR SOCCER TEAM"
120 PRINT "RATE EACH OF THE 6 PLAYERS YOU THINK ARE
    THE BEST"
130 PRINT "ON THE FOLLOWING CATEGORIES 1 - 5 (5 BEST)"
```

```

140 PRINT TAB( 10), "SOCCER SKILLS"
150 PRINT TAB( 10), "ATTITUDE"
160 PRINT TAB( 10), "ATHLETIC ABILITY"
170 PRINT "THE COMPUTER WILL CALCULATE A MVP SCORE"
180 PRINT
190 FOR I = 1 TO 6
200 PRINT "PLAYER "; I;
210 INPUT S(I), A(I), G(I)
220 NEXT I
230 FOR I = 1 TO 6
240 T(I) = S(I) * 3 + A(I) + G(I) * 2
250 NEXT I
260 PRINT "THE RATINGS ARE AS FOLLOWS"
265 PRINT
270 PRINT "PLAYER SKILL ATTITUDE ABILITY MVP"
300 FOR I = 1 TO 6
310 PRINT " "; I; " "; S(I); " "; A(I); "
"; G(I); " "; T(I)
320 NEXT I
350 END

```

**Note:** Line 95: The DIM statement is used for numeric variables too. It saves space for you to store more than one number under the same name. Then, you use subscripts to get to each number.

Line 210: The ratings for player 1 are put in S(1), A(1) and G(1). The ratings for player 2 are in S(2), etc.

Line 240: Another name for a subscripted variable is an array.

## VARIABLE LIST

S—Soccer skills ratings  
A—Attitude ratings  
G—Athletic ability ratings  
T—MVP score

## COMPUTER NOTES

The spacing produced by the , (comma) and the ; (semicolon) varies with each version of BASIC. Therefore, this program may not

line up on your computer as shown in the sample output. Experiment with the spaces used in the print statements in order to fix it for your computer.

**Atari BASIC** Does not allow subscripted variables in an input statement. To convert, replace line 210 with:

```
205 INPUT X, Y, Z
```

```
206 LET S(I) = X
```

```
207 LET A(I) = Y
```

```
210 LET G(I) = Z
```

---

# BANK STATEMENT

---

Now you can print your own bank statement. Use this program to show how your money increased (or decreased) during the last few months.

## SAMPLE RUN

BANKER'S PROGRAM - PRINTS A BANK STATEMENT  
INPUT BEGINNING BALANCE?100  
INPUT TOTAL NUMBER OF DEPOSITS AND WITHDRAWALS?4  
NOW INPUT EACH DEPOSIT OR WITHDRAWAL  
WITHDRAWALS ARE (-)  
AMOUNT1?200  
AMOUNT2?300  
AMOUNT3?-500  
AMOUNT4?100

\*\*\*\* BANK STATEMENT \*\*\*\*

BEGINNING BALANCE \$100

#	AMOUNT	ACCOUNT BALANCE
1	200	300
2	300	600
3	-500	100
4	100	200

ENDING BALANCE \$200

## PROGRAM LISTING

```
100 DIM T(15),D(14)
110 PRINT "BANKER'S PROGRAM - PRINTS A BANK STATEMENT"
120 PRINT "INPUT BEGINNING BALANCE";
130 INPUT A
140 PRINT "INPUT TOTAL NUMBER OF DEPOSITS AND
WITHDRAWALS";
```

```

150 INPUT N
160 PRINT "NOW INPUT EACH DEPOSIT OR WITHDRAWAL"
170 PRINT "WITHDRAWALS ARE (-)"
180 FOR I = 1 TO N
190 PRINT "AMOUNT"; I;
200 INPUT D(I)
210 NEXT I
220 T(1) = A
230 FOR I = 1 TO N
240 T(I + 1) = T(I) + D(I)
250 NEXT I
260 PRINT
261 PRINT
270 PRINT TAB(9); "**** BANK STATEMENT ****"
280 PRINT
281 PRINT
290 PRINT TAB(13); "BEGINNING BALANCE $"; A
300 PRINT
301 PRINT
310 PRINT "#", "AMOUNT", "ACCOUNT"
320 PRINT TAB(29); "BALANCE"
323 PRINT
330 FOR I = 1 TO N
340 PRINT I, D(I), T(I+1)
350 NEXT I
360 PRINT
361 PRINT
370 PRINT TAB(15); "ENDING BALANCE $"; T(N + 1)
380 END

```

Note: Line 240: Try playing computer, and do this calculation yourself. Why does the program store the result in T (I + 1)?

### VARIABLE LIST

A—Beginning balance  
D—Deposits/withdrawals  
T—Running balance

### COMPUTER NOTES

Atari-BASIC      Replace line 200 with:  
                    200 Input X  
                    205 D(I) = X

---

# PASCAL'S TRIANGLE

---

The numbers in this program have an interesting relationship. Can you figure out what it is? This is called Pascal's triangle, named after the mathematician who invented this interesting arrangement of numbers.

## SAMPLE RUN

HERE IS PASCAL'S TRIANGLE

```

          0
         1 1
        1 2 1
       1 3 3 1
      1 4 6 4 1
     1 5 10 10 5 1
    1 6 15 20 15 6 1
   1 7 21 35 35 21 7 1
  1 8 28 56 70 56 28 8 1
 1 9 36 84 126 126 84 36 9 1
```

## PROGRAM LISTING

```
100 PRINT "HERE IS PASCAL'S TRIANGLE"
110 DIM E[10,10]
120 LET M=35
130 REM<<PUT NUMBERS INTO ARRAY
140 LET E[1,1]=0
150 LET N=6
160 FOR R=2 TO N
170 E[R,1]=1
180 FOR S=2 TO R-1
190 E[R,S]=E[R-1,S-1]+E[R-1,S]
200 NEXT S
205 E[R,R]=1
210 NEXT R
220 REM<<NOW PRINT TRIANGLE
240 FOR I=1 TO N
250 PRINT TAB(M-(I-1)*6);
260 FOR J=1 TO I
270 PRINT E[I,J]; " ";
280 NEXT J
290 PRINT
300 NEXT I
310 END
```

**Note:** Line 190: This statement calculates each row of numbers. Notice that you can add together numbers in the same array.  
Line 110: The array E has 2 subscripts. This is called a 2 dimensional array. A 2 dimensional array has rows and columns. In this case, the array E has 10 rows and 10 columns.

### VARIABLE LIST

**E**—Contains the numbers for Pascal's Triangle

**N**—Number of rows to print

---

# SEA BATTLE

---

In this program you and the computer play the Sea Battle game. Deploy your ships on a 4 x 4 grid with each ship occupying one square, and the computer will do the same. Then fire away!

If this gets too tame for you, see if you can change the program to increase the size of the grid or the number of ships.

## SAMPLE RUN

### BATTLESHIP

WOULD YOU LIKE INSTRUCTIONS(Y/N)?Y  
YOU HAVE A FLEET OF 6 SHIPS. EACH SHIP IS NUMBERED 1,  
2, OR 3 (2 OF EACH NUMBER). USING A 4 X 4 GRID  
YOU AND THE COMPUTER FIRE TORPEDOES AT EACH OTHER.  
WHEN YOU ARE ASKED, INPUT YOUR SHOT(SUCH AS 2,3  
STAYING WITHIN 4). WHEN THE COMPUTER SHOOT, IF  
IT HITS A SHIP, INPUT WHICH SHIP IT HIT. WHEN THE COMPUTER  
MISSES, INPUT A 4. THE FIRST ONE TO DESTROY THEIR OPPONENT'S  
FLEET, WINS. GOOD LUCK!!!

SHOOTING AT 2 , 1

?4

I HAVE A SCORE OF 0

NOW YOU TRY - INPUT YOUR SHOT?2,3

YOU MISSED

SHOOTING AT 1 , 3

?4

I HAVE A SCORE OF 0

NOW YOU TRY - INPUT YOUR SHOT?1,1

◆ ◆ ◆ ◆ B O O M ◆ ◆ ◆ ◆

YOU HIT A 3

YOUR SCORE IS 3

SHOOTING AT 2 , 2

?3

◆ ◆ ◆ ◆ B A N G ◆ ◆ ◆ ◆

I HAVE A SCORE OF 3

NOW YOU TRY - INPUT YOUR SHOT?2,2

YOU MISSED

SHOOTING AT 3 , 1

74

I HAVE A SCORE OF 3  
NOW YOU TRY - INPUT YOUR SHOT?3,3

◆ ◆ ◆ ◆ B O O M ◆ ◆ ◆ ◆  
YOU HIT A 1  
YOUR SCORE IS 4

### PROGRAM LISTING

```
100 DIM US[1]
105 REM◆◆THIS PROGRAM SIMULATES A GAME OF BATTLESHIP◆◆
110 PRINT TAB(27)"BATTLESHIP"
115 PRINT
120 PRINT
125 PRINT
130 PRINT "WOULD YOU LIKE INSTRUCTIONS(Y/N)";
135 INPUT US
140 IF US="N" THEN 205
145 PRINT "YOU HAVE A FLEET OF 6 SHIPS. EACH SHIP IS
    NUMBERED 1,"
150 PRINT "2, OR 3(2 OF EACH NUMBER). USING A 4 X 4 GRID"
155 PRINT "YOU AND THE COMPUTER FIRE TORPEDOES AT EACH
    OTHER."
160 PRINT "WHEN YOU ARE ASKED, INPUT YOUR SHOT(SUCH AS
    2,3"
165 PRINT "STAYING WITHIN 4). WHEN THE COMPUTER SHOOTS,
    IF"
170 PRINT "IT HITS A SHIP, INPUT WHICH SHIP IT HIT.
    WHEN THE COMPUTER"
175 PRINT "MISSES, INPUT A 4. THE FIRST ONE TO DESTROY
    THEIR OPPONENT'S"
180 PRINT "FLEET, WINS. GOOD LUCK!!!"
185 PRINT
190 PRINT
195 PRINT
200 REM◆◆SET SIZE OF GRID◆◆
205 DIM C[4,4],P[4,4]
210 DIM T[2]
215 REM◆◆SET ARRAYS TO ZERO◆◆
220 FOR I=1 TO 4
225 FOR J=1 TO 4
230 C[I,J]=0
235 P[I,J]=0
240 NEXT J
245 NEXT I
250 T[1]=0
255 T[2]=0
```

```

260 REM♦♦SET L TO SIZE OF GRID♦♦
265 L=4
270 M=0
275 REM♦♦SET UP COMPUTER'S SHIPS ON L X L GRID♦♦
280 FOR J=1 TO 2
285 FOR I=1 TO 3
290 R=INT(RND(1)*L)+1
295 S=INT(RND(1)*L)+1
300 C[R,S]=I
305 NEXT I
310 NEXT J
315 REM♦♦GET COMPUTER'S TARGET♦♦
320 M=M+1
325 IF M>L*L THEN 465
330 R=INT(RND(1)*L)+1
335 S=INT(RND(1)*L)+1
340 IF P[R,S]>0 THEN 330
345 PRINT "SHOOTING AT ";S,"";R
350 INPUT H
355 P[R,S]=H
360 IF H=4 THEN 380
365 PRINT "♦ ♦ ♦ ♦ B A N G ♦
♦ ♦ ♦ ♦"
370 T[1]=T[1]+H
375 IF T[1]>11 THEN 465
380 PRINT "I HAVE A SCORE OF ";T[1]
385 PRINT "NOW YOU TRY - INPUT YOUR SHOT";
390 INPUT R,S
395 IF C[R,S]=5 THEN 415
400 IF C[R,S]>0 THEN 425
405 PRINT "YOU MISSED"
410 GOTO 320
415 PRINT "SPLASH! YOU ALREADY SHOT THERE!!!"
420 GOTO 320
425 PRINT CHR$(7)
430 PRINT "♦ ♦ ♦ ♦ B O O M ♦
♦ ♦ ♦ ♦"
435 PRINT "YOU HIT A ";C[R,S]
440 T[2]=T[2]+C[R,S]
445 PRINT "YOUR SCORE IS ";T[2]
450 C[R,S]=5
455 IF T[2]>11 THEN 465
460 GOTO 320
465 PRINT TAB(23)"GAME OVER"
470 IF T[2]>T[1] THEN 485
475 PRINT "!!!COMPUTER WINS!!!"
480 GOTO 490
485 PRINT "!!!YOU WIN!!!"
490 PRINT

```

```
495 PRINT
500 PRINT
505 PRINT "WANT A NEW FLEET AND A NEW GAME(Y/N)";
510 INPUT US
515 IF US="Y" THEN 115
520 PRINT
525 PRINT "BETTER LUCK NEXT TIME!"
530 PRINT
535 END
```

**Note:** Line 130: When the instructions to a game are long, this statement gives the player the option of skipping the instructions.

**Line 205:** C and P are called 2 dimensional arrays. These arrays look like the grid you use to play this game. Each number occupies one cell in the grid.

**Line 355:** Each time the computer shoots, the result of the shot is put in the player's array. This keeps the computer from shooting at the same place twice.

### VARIABLE LIST

C—Computer's ships on the 4 x 4 grid

P—Player's ships on the 4 x 4 grid

T—Score for each side

---

# 11 SUBROUTINES

## How the Pros Do It

---

Subroutines are small programs that a larger program uses over and over. Instead of including them every time you need them, they only appear once near the end of a program. A GOSUB statement in the large program tells the computer to leave that program and go to the subroutine. When the subroutine is finished, a RETURN statement tells the computer to go back to where it came from in the main program.

### PROBLEMS

1. Birthday
2. Math Quiz
3. Tennis

---

# BIRTHDAY

---

How many days before your birthday? How many days between now and Christmas? This program can tell you. It converts each date into a Julian date and then calculates the difference. The Julian date is the number of days since some predetermined date, and in this program, that date is 1960. Notice that the program has to add extra days for leap years.

## SAMPLE RUN

USE THIS PROGRAM TO FIND OUT HOW MANY DAYS UNTIL YOUR BIRTHDAY

INPUT TODAY'S DATE AS FOLLOWS

YEAR?1981

MONTH(1-12)?11

DAY?4

INPUT DATE OF YOUR NEXT BIRTHDAY

YEAR?1982

MONTH(1-12)?4

DAY?1

THERE ARE 147 DAYS UNTIL YOUR BIRTHDAY AGAIN

?Y

INPUT DATE OF YOUR NEXT BIRTHDAY

YEAR?1981

MONTH(1-12)?11

DAY?30

THERE ARE 26 DAYS UNTIL YOUR BIRTHDAY AGAIN

?N

## PROGRAM LISTING

```
100 DIM A$(1),D(12)
105 PRINT "USE THIS PROGRAM TO FIND OUT HOW MANY DAYS
      UNTIL YOUR BIRTHDAY"
110 PRINT "INPUT TODAY'S DATE AS FOLLOWS"
115 GOSUB 195
120 GOSUB 270
125 REM NOW FIND OUT HOW MANY DAYS SINCE 1960
130 GOSUB 310
135 C=T
```

```

140 PRINT "INPUT DATE OF YOUR NEXT BIRTHDAY"
145 GOSUB 270
150 GOSUB 310
155 N=T-C
160 PRINT "THERE ARE ";N;" DAYS UNTIL YOUR BIRTHDAY"
165 PRINT "AGAIN"
170 INPUT AS
175 IF AS="Y" THEN 140
180 GOTO 360
185 REM◆◆ SUBROUTINE
190 REM◆◆ SETS D=TO NUMBER OF DAYS IN EACH MONTH
195 LET D[1]=31
200 LET D[2]=28
205 LET D[3]=31
210 LET D[4]=30
215 LET D[5]=31
220 LET D[6]=30
225 LET D[7]=31
230 LET D[8]=31
235 LET D[9]=30
240 LET D[10]=31
245 LET D[11]=30
250 LET D[12]=31
255 RETURN
260 REM◆◆ SUBROUTINE
265 REM◆◆ INPUTS YEAR MONTH AND DAY
270 PRINT "YEAR";
275 INPUT Y
280 PRINT "MONTH(1-12)";
285 INPUT M
290 PRINT "DAY";
295 INPUT A
300 RETURN
305 REM◆◆ SUBROUTINE
310 REM◆◆ CALCULATES NUMBER OF DAYS SINCE 1960
315 T=Y-1960
320 L=INT(T/4)
325 T=(T-L)*365+L*366
330 FOR I=1 TO M-1
335 T=T+D[I]
340 NEXT I
345 REM◆◆ ADD IN DAYS OF THIS MONTH
350 T=T+A
355 RETURN
360 END

```

Note: Line 115: This GOSUB statement tells the computer to execute statement number 195 next.

Line 255: The RETURN statement tells the computer to go back and execute the line after the GOSUB statement. Subroutines are useful when you want to do the same calculation with different numbers.

Lines 315-320: This statement figures out how many leap years since 1960.

Lines 330-340: Add in all the days of each month of the current year.

Lines 195-250: Since the number of days in each month won't change, it is easier to put these values into the program. Otherwise, you would have to input them every time.

### VARIABLE LIST

D—Array containing the number of days for each month in the year

Y—Year

M—Month

A—Day

T—Number of days since 1960 for today's date

C—Number of days since 1960 for next birthday

N—Number of days between T and C

L—Number of leap years since 1960

---

# MATH QUIZ

---

Here is a chance to test your math skills on the computer. You can do addition, subtraction, multiplication and division. The computer gives you the problem, and you input the answer.

Can you change this program so it keeps track of the number of correct answers?

## SAMPLE RUN

```
INPUT MATH OPERATION YOU WANT TO TRY
+ ADDITION
- SUBTRACTION
* MULTIPLICATION
/ DIVISION
X STOP
TO CHANGE OPERATIONS TYPE -1 AS A ANSWER
INPUT OPERATION SYMBOL?+
INPUT LIMITS OF YOUR NUMBERS?1,10
INPUT NUMBER OF INCORRECT RESPONSES?2
  5   + 3   =?8
YOU ARE RIGHT!
  5   + 2   =?6
SORRY . . . TRY AGAIN?5
THAT IS 2   TRIES
THE CORRECT ANSWER IS           7
  4   + 1   =?-1
INPUT OPERATION SYMBOL?-
INPUT LIMITS OF YOUR NUMBERS?1,10
INPUT NUMBER OF INCORRECT RESPONSES?2
  6   - 1   =?5
YOU ARE RIGHT!
  3   - 1   =?2
YOU ARE RIGHT!
  7   - 5   =?-1
INPUT OPERATION SYMBOL?X
```

## PROGRAM LISTING

```
100 DIM O$(10),A$(1)
105 LET O$="+-*/X"
110 PRINT "INPUT MATH OPERATION YOU WANT TO TRY"
115 PRINT "+ ADDITION"
120 PRINT "- SUBTRACTION"
```

```

125 PRINT "* MULTIPLICATION"
130 PRINT "/ DIVISION"
135 PRINT "X STOP"
140 PRINT "TO CHANGE OPERATIONS TYPE -1 AS AN ANSWER"
145 PRINT "INPUT OPERATION SYMBOL ";
150 INPUT A$
155 REM ***FIND OUT WHAT OPERATION NUMBER
160 GOSUB 505
165 PRINT "INPUT LIMITS OF YOUR NUMBERS(L,H)"
170 INPUT J,K
175 PRINT "INPUT NUMBER OF INCORRECT RESPONSES";
180 INPUT T
185 REM ***GET 2 RANDOM NUMBERS M AND N
190 GOSUB 370
195 I=0
200 ON A GOTO 210,245,285,325
205 REM ***ADDITION***
210 L=M+N
215 PRINT M; " + " ; N ; " = ";
220 INPUT P
225 IF P<0 THEN 145
230 IF P=L THEN 435
235 GOSUB 455
240 GOTO 220
245 REM ** SUBT
250 L=M-N
255 PRINT M; " - "; N; " = ";
260 INPUT P
265 IF P<0 THEN 145
270 IF P=L THEN 435
275 GOSUB 455
280 GOTO 260
285 REM ** MULT
290 L=M*N
295 PRINT M; " * "; N; " = ";
300 INPUT P
305 IF P<0 THEN 145
310 IF P=L THEN 435
315 GOSUB 455
320 GOTO 300
325 REM ** DIV
330 L=M/N
335 PRINT M; " / "; N; " = ";
340 INPUT P
341 IF E>=T THEN 305
342 PRINT
343 IF W=6 THEN 250
345 IF P<0 THEN 145
350 IF P=L THEN 435

```

```

355 GOSUB 455
360 GOTO 300
365 REM ***SUBROUTINE
370 REM ***FINDS 2 RANDOM NUMBERS WITHIN
    THE LIMITS OF J & K
375 M=INT(RND(1)*K+1)
380 IF M<J THEN 375
385 N=INT(RND(1)*K+1)
390 IF N<J THEN 385
395 IF A=1 THEN 425
400 IF A=3 THEN 425
405 IF M<=N THEN 375
415 IF INT(M/N)*N=(M/N)*N THEN 425
420 GOTO 375
425 RETURN
430 REM ***SUBROUTINE
435 REM ***PRINTS IF THE RESPONSE IS CORRECT
440 PRINT "YOU ARE RIGHT!"
445 GOTO 190
450 REM ***SUBROUTINE
455 REM ***PRINTS IF THE RESPONSE WAS WRONG
460 I=I+1
465 IF I>=T THEN 480
470 PRINT "SORRY . . . TRY AGAIN";
475 RETURN
480 PRINT "THAT IS ";I;" TRIES"
485 PRINT "THE CORRECT ANSWER IS ";L
490 GOTO 190
495 REM ***SUBROUTINE
500 REM ***GETS THE OPERATION NUMBER FROM A$
505 FOR A=1 TO 5
510 IF A$=MID$(O$,A,1) THEN 530
515 NEXT A
520 PRINT "TYPE +,-,*, OR /"
525 GOTO 145
530 IF A=5 THEN 540
535 RETURN
540 END

```

Note: Line 365: It is smart to set your subroutines off with REM statements. Then you can find them quickly if you are changing your program.

Lines 415-420: These statements force the result of M/N to be a whole number.

## VARIABLE LIST

- A\$ —Math operation code
- J —Lower limit of numbers
- K —Upper limit of numbers
- T —Number of incorrect responses
- I —Number of responses
- A —Operation number
- M,N —Numbers used in calculation
- L —Result of calculation
- P —User's response

## COMPUTER NOTES

- |               |  |
|---------------|--|
| Atari-BASIC   | Replace line 510 with:<br>510 IF A\$ = O\$(A,A) THEN 530       |
| HP-2000 BASIC | Replace line 510 with:<br>510 IF A\$ = O\$(A,A) THEN 530       |
| TI-BASIC      | Replace line 510 with:<br>510 IF A\$ = SEG\$(O\$,A,1) THEN 530 |

---

# TENNIS

---

Try playing tennis with the computer. Each time the ball is hit to you, type in a number from 1 to 4. Watch out! the computer has been taking lessons.

Try changing the program to keep track of the set score.

Can you add other shots to the game or change it so the rally can go longer?

## SAMPLE RUN

DO YOU WANT INSTRUCTIONS?

?Y

YOU ARE PLAYING /NO ADD/ TENNIS WITH THE COMPUTER.  
THE FIRST PLAYER TO GET 4 POINTS WINS THE GAME.  
OK...THE COMPUTER IS SPINNING THE RACQUET  
THE COMPUTER WINS THE SPIN AND ELECTS TO SERVE.  
READY...BEGIN!!

COMPUTER SERVES SMASH TO BACKHAND

YOU -INPUT YOUR SHOT(1-4) ?1

HIT SOFT BALL SHORT

COMPUTER LOB HIGH TO FOREHAND

YOU -INPUT YOUR SHOT(1-4) ?3

HIT BLAZING FOREHAND TO WIN POINT

SERVER: 0

OPPONENT: 1

COMPUTER SERVES SMASH TO FOREHAND

YOU -INPUT YOUR SHOT(1-4) ?2

HIT SOFT BALL SHORT

COMPUTER HIT BALL OUT OF COURT

SERVER: 0

OPPONENT: 2

COMPUTER SERVES FAULT

COMPUTER SERVES FAULT

DOUBLE FAULT

SERVER: 0

OPPONENT: 3

COMPUTER SERVES HIT SOFT SHOT

YOU -INPUT YOUR SHOT(1-4) ?1

HIT BALL OVER FENCE

SERVER: 1  
OPPONENT: 3

COMPUTER SERVES HIT SOFT SHOT  
YOU -INPUT YOUR SHOT(1-4) ?2  
HIT BALL DEEP TO BACKHAND  
COMPUTER LOB HIGH TO FOREHAND  
YOU -INPUT YOUR SHOT(1-4) ?1  
SMASH OVERHEAD INTO NEXT COURT  
SERVER: 2  
OPPONENT: 3

### PROGRAM LISTING

```
95  DIM A$(1)
100 PRINT "TENNIS"
101 PRINT "DO YOU WANT INSTRUCTIONS?"
102 INPUT A$
103 IF A$ = "N" THEN 105
104 GOSUB 650
105 DIM C(5)
106 T = 1
107 O = 0
108 S = 0
110 PRINT
111 F = 0
115 ON T GOTO 120,180
120 PRINT "COMPUTER SERVES ";
125 R = INT ( RND (1) * 5 + 1)
130 GOSUB 340
131 IF N = 0 THEN 135
132 IF N = 1 THEN 115
133 IF N = 2 THEN 275
135 PRINT "YOU ";
140 GOSUB 230
145 GOSUB 425
146 IF N = 1 THEN 275
150 PRINT "COMPUTER ";
155 R = INT ( RND (1) * 5 + 1)
160 GOSUB 495
161 IF N = 1 THEN 275
165 PRINT "YOU ";
170 GOSUB 230
175 GOSUB 565
176 GOTO 275
180 PRINT "YOU SERVE ";
185 GOSUB 230
190 GOSUB 340
```

```

191 IF N = 0 THEN 195
192 IF N = 1 THEN 115
193 IF N = 2 THEN 275
195 PRINT "COMPUTER ";
196 R = INT ( RND (1) * 5 + 1)
200 GOSUB 425
201 IF N = 1 THEN 275
205 PRINT "YOU ";
210 GOSUB 230
215 GOSUB 495
216 IF N = 1 THEN 275
220 PRINT "COMPUTER ";
221 R = INT ( RND (1) * 5 + 1)
225 GOSUB 565
226 GOTO 275
229 REM **SUBROUTINE
230 REM ** GETS PLAYER'S SHOT
235 PRINT "-INPUT YOUR SHOT(1-4) ";
240 INPUT B
241 IF B > 4 THEN 243
242 GOTO 245
243 PRINT "INPUT A NUMBER FROM 1 TO 4"
244 GOTO 240
245 FOR I = 1 TO 4
250 J = INT ( RND (1) * 5 + 1)
255 C(I) = J
260 NEXT I
265 R = C(B)
270 RETURN
275 REM ** PRINTS SCORE
280 PRINT "SERVER: ";S
285 PRINT "OPPONENT: ";O
286 PRINT
290 IF S = 4 THEN 305
295 IF O = 4 THEN 315
300 GOTO 111
305 PRINT "SERVER WINS"
310 GOTO 320
315 PRINT "OPPONENT WINS"
320 IF T = 1 THEN 330
325 GOTO 106
330 T = 2
335 GOTO 107
339 REM ** SUBROUTINE
340 REM ** RESULT OF SERVE
341 N = 0
345 ON R GOTO 350,360,370,405,415
350 PRINT "SMASH TO BACKHAND"
355 RETURN

```

```

360 PRINT "SMASH TO FOREHAND"
365 RETURN
370 PRINT "FAULT"
375 F = F + 1
380 IF F = 2 THEN 390
384 N = 1
385 RETURN
390 PRINT "DOUBLE FAULT"
395 O = O + 1
399 N = 2
400 RETURN
405 PRINT "HIT SOFT SHOT"
410 RETURN
415 PRINT "FAULT INTO NET"
420 GOTO 375
424 REM ** SUBROUTINE
425 REM ** RESULT OF RETURN
426 N = 0
430 ON R GOTO 435,445,455,470,485
435 PRINT "HIT BALL LOW TO FOREHAND"
440 RETURN
445 PRINT "HIT SOFT BALL SHORT"
450 RETURN
455 PRINT "HIT BALL INTO NET"
460 S = S + 1
461 N = 1
465 RETURN
470 PRINT "HIT BALL OVER FENCE"
475 S = S + 1
479 N = 1
480 RETURN
485 PRINT "HIT BALL DEEP TO BACKHAND"
490 RETURN
494 REM ** SUBROUTINE
495 REM ** SECOND ROUND OF HITS
499 N = 0
500 ON R GOTO 505,520,530,540,550
505 PRINT "RUSH NET AND MISS VOLLEY"
510 O = O + 1
514 N = 1
515 RETURN
520 PRINT "SMASH OVERHEAD DEEP TO BACKHAND"
525 RETURN
530 PRINT "HIT FOREHAND SHORT"
535 RETURN
540 PRINT "LOB HIGH TO FOREHAND"
545 RETURN
550 PRINT "HIT BALL OUT OF COURT"
551 N = 1

```

```

555 O = O + 1
560 RETURN
564 REM ** SUBROUTINE
565 REM ** FINAL POINT
570 ON R GOTO 575,590,605,620,635
575 PRINT "HIT WINNER DOWN LINE"
580 O = O + 1
585 RETURN
590 PRINT "FALL DOWN SLIDING FOR A VOLLEY"
595 S = S + 1
600 RETURN
605 PRINT "SMASH OVERHEAD INTO NEXT COURT"
610 S = S + 1
615 RETURN
620 PRINT "HIT BLAZING FOREHAND TO WIN POINT"
625 O = O + 1
630 RETURN
635 PRINT "RUN INTO FENCE TRYING TO HIT BALL"
640 S = S + 1
645 RETURN
650 REM **INSTRUCTION SUBROUTINE
655 PRINT "YOU ARE PLAYING /NO ADD/ TENNIS WITH THE
COMPUTER."
660 PRINT "THE FIRST PLAYER TO GET 4 POINTS WINS THE
GAME."
665 PRINT "OK...THE COMPUTER IS SPINNING THE RACQUET"
670 PRINT "THE COMPUTER WINS THE SPIN AND ELECTS TO
SERVE."
675 PRINT "READY...BEGIN!!"
680 RETURN
685 END

```

**Note:** Lines 240-265: This subroutine uses a little trick in order to get a random number for the player's response.

Notice how each subroutine does one task: that is, a subroutine gets the result of a serve, or result of a return. This makes it easier for you to add to or change the program later on.

### VARIABLE LIST

T—Determines who serves  
 T = 1 Computer serves  
 T = 2 Player serves

- S—Score for server
- O—Score for receiver of serve
- F—Number of faults from serve
- R—Random number determining result of hit
- B—Player's input
- A—Result of player's shot
- N—Determines when point is over

# 12

---

# GAMES

## Now You Can Do Your Own Thing

---

Now you are ready to write your own computer games. Most computer games contain a set of instructions, some player options, random consequences and a scoring system. Try these games first and then make up one of your own.

### PROBLEMS

1. Ski Weekend
2. Backpack

---

# SKI WEEKEND

---

Your parents say you may join your friends for a ski weekend in 6 weeks if you earn enough money for the trip. Your share of the weekend will cost \$145. See if you can make it.

You can change the program to a game saving money for a ten speed, some camping equipment or something else special. Try changing the jobs also.

## SAMPLE RUN

WANT INSTRUCTIONS? ?Y

YOU HAVE 6 WEEKS TO EARN \$145 FOR A SKI WEEKEND

YOU HAVE A CHOICE OF 3 JOBS

1. PAPER ROUTE - YOU CAN EARN \$25 PER WEEK
2. BABYSITTING - YOU CAN EARN \$1.25 PER HOUR
3. CAR WASH - YOU CAN EARN \$2.00 PER CAR

INPUT THE JOB NUMBER 1 - 3 TO CHOOSE A JOB

IF YOU WORK HARD AND HAVE A LITTLE LUCK

YOU WILL HIT THE SLOPES IN 6 WEEKS

HAPPY HOLIDAYS

CHOOSE JOB ?2

2. BABYSITTING - YOU CAN EARN \$1.25 PER HOUR

WEEK 1

THE PARENTS LOVE YOU

ADD \$4.00 IN TIPS

YOU HAVE EARNED \$ 29 SO FAR

CHANGE JOBS ?N

WEEK 2

THE PARENTS LOVE YOU

ADD \$4.00 IN TIPS

YOU HAVE EARNED \$ 58 SO FAR

CHANGE JOBS ?

WEEK 3

THE CHILD HAS A FRIEND OVER

YOUR WAGES ARE INCREASED TO \$31.50  
YOU HAVE EARNED \$ 89.5 SO FAR

CHANGE JOBS ?N

WEEK 4

YOUR EXTRA-CURRICULAR ACTIVITIES INTERFERE WITH YOUR JOB  
YOU LOSE \$8.00 IN EXPECTED INCOME  
CHOOSE ANOTHER JOB  
YOU HAVE EARNED \$ 106.5 SO FAR

CHOOSE JOB ?1

1. PAPER ROUTE - YOU CAN EARN \$25 PER WEEK

WEEK 5

YOU HAVE SEVERAL HEAVY TIPPERS WHEN YOU COLLECT  
INCREASE YOUR WAGES TO 29  
YOU HAVE EARNED \$ 135.5 SO FAR

CHANGE JOBS ?N

WEEK 6

RAIN RUINS HALF OF YOUR PAPERS  
YOU LOSE HALF OF YOUR WEEK'S EARNINGS  
YOU HAVE EARNED \$ 148 SO FAR

HAPPY HOLIDAYS!!!

YOU EARNED \$ 148 IN 6 WEEKS

WANT TO TRY AGAIN?

?Y

CHOOSE JOB ?3

3. CAR WASH - YOU CAN EARN \$2.00 PER CAR

WEEK 1

YOU GET A FRIEND TO HELP YOU  
TOGETHER YOU DO 32 CARS. YOUR SHARE IS \$32.00  
YOU HAVE EARNED \$ 32 SO FAR

CHANGE JOBS ?N

WEEK 2

YOU GET A FRIEND TO HELP YOU  
TOGETHER YOU DO 32 CARS. YOUR SHARE IS \$32.00  
YOU HAVE EARNED \$ 64 SO FAR

## PROGRAM LISTING

```
95  DIM A$(1)
100 PRINT "WANT INSTRUCTIONS? ";
110 INPUT A$
120 IF A$ = "Y" THEN 800
130 LET T = 145
140 LET W = 0
145 E = 0
150 PRINT "CHOOSE JOB ";
160 INPUT J
165 GOSUB 400
170 REM WEEK NUMBER
180 W = W + 1
185 PRINT
200 PRINT "WEEK ";W
210 ON J GOTO 500,600,700
250 PRINT "YOU DIDN'T MAKE IT - SORRY"
260 PRINT "MAYBE NEXT YEAR"
270 PRINT "WANT TO TRY AGAIN? "
280 INPUT A$
290 IF A$ = "Y" THEN 130
300 GOTO 999
305 PRINT
306 PRINT "HAPPY HOLIDAYS!!!"
310 PRINT "YOU EARNED $ ";E;" IN ";W;" WEEKS"
315 PRINT
320 GOTO 270
340 PRINT "YOU HAVE EARNED $ ";E;" SO FAR"
341 IF E > = T THEN 305
342 PRINT
343 IF W = 6 THEN 250
345 IF F = 1 THEN 150
350 PRINT "CHANGE JOBS ";
360 INPUT A$
370 IF A$ = "Y" THEN 150
380 GOTO 180
395 REM *** SUBROUTINE
400 REM ** DESCRIBES JOBS
410 ON J GOTO 415,430,445
415 PRINT "1. PAPER ROUTE - YOU CAN EARN $25 PER WEEK"
420 RETURN
430 PRINT "2. BABYSITTING - YOU CAN EARN $1.25 PER
    HOUR"
441 RETURN
445 PRINT "3. CAR WASH - YOU CAN EARN $2.00 PER CAR"
455 RETURN
500 REM ** PAPER ROUTE
510 R = INT ( RND (1) * 6 + 1)
```

```

515 F = 0
520 ON R GOTO 525,545,560,575,585,592
525 PRINT "YOU SELL 10 EXTRA PAPERS THIS WEEK"
526 PRINT "YOUR WAGES INCREASE TO $28.50"
530 E = E + 28.5
540 GOTO 340
545 PRINT "YOU HAVE SEVERAL HEAVY TIPPERS WHEN YOU
COLLECT"
546 PRINT "INCREASE YOUR WAGES TO "29.00"
550 E = E + 29
555 GOTO 340
560 PRINT "YOU WIN A CASH BONUS FROM THE PAPER"
561 PRINT "ADD AN EXTRA $25.00 TO YOUR EARNINGS"
565 E = E + 50
570 GOTO 340
575 PRINT "YOU ARE WAIT-LISTED FOR A ROUTE"
576 PRINT "LOSE 1 WEEK'S SALARY"
580 F = 1
581 GOTO 340
585 PRINT "RAIN RUINS HALF OF YOUR PAPERS"
586 PRINT "YOU LOSE HALF OF YOUR WEEK'S EARNINGS"
590 E = E + 12.5
591 GOTO 340
592 PRINT "THE NEWSPAPER FOLDS AND YOU ARE OUT OF WORK"
593 PRINT "YOU GET HALF A WEEK'S SALARY - CHOOSE
ANOTHER JOB"
595 F = 1
599 GOTO 340
600 REM ** BABYSITTING
601 R = INT ( RND (1) * 6 + 1)
602 F = 0
603 ON R GOTO 605,615,630,645,660,675
605 PRINT "THE CHILD GETS SICK AND YOU ONLY WORK HALF
A WEEK"
610 PRINT "YOU EARN $12.5 AND MUST CHOOSE ANOTHER JOB"
611 E = E + 12.5
612 F = 1
613 GOTO 340
615 PRINT "THE PARENTS AREN'T SATISFIED WITH YOUR WORK"
616 PRINT "AND HIRE SOMEONE ELSE - CHOOSE ANOTHER JOB"
617 PRINT "YOU ONLY EARN $10.00"
620 E = E + 10
622 F = 1
625 GOTO 340
630 PRINT "YOUR EXTRA-CURRICULAR ACTIVITIES INTERFERE
WITH YOUR JOB"
631 PRINT "YOU LOSE $8.00 IN EXPECTED INCOME"
632 PRINT "CHOOSE ANOTHER JOB"
633 F = 1

```

```

635 E = E + 17
640 GOTO 340
645 PRINT "THE PARENTS LOVE YOU"
646 PRINT "ADD $4.00 IN TIPS"
650 E = E + 29
655 GOTO 340
660 PRINT "THE CHILD HAS A FRIEND OVER"
661 PRINT "YOUR WAGES ARE INCREASED TO $31.50"
665 E = E + 31.5
670 GOTO 340
675 PRINT "THE PARENTS WERE GONE LATER THAN PLANNED"
676 PRINT "YOUR WAGES ARE $30.00"
680 E = E + 30
685 GOTO 340
700 REM ** CAR WASH
701 R = INT ( RND (1) * 6 + 1)
705 F = 0
706 ON R GOTO 710,720,730,745,765,780
710 PRINT "YOU GET SOME BIG TIPS"
711 PRINT "YOUR WAGES RISE TO $35.00"
715 E = E + 35
716 GOTO 340
720 PRINT "YOU HAVE LOTS OF CUSTOMERS"
721 PRINT "YOUR EARNINGS ARE $38.00"
725 E = E + 38
726 GOTO 340
730 PRINT "YOU GET A FRIEND TO HELP YOU"
731 PRINT "TOGETHER YOU DO 32 CARS. YOUR SHARE IS
      $32.00"
735 E = E + 32
740 GOTO 340
745 PRINT "IT RAINS AND YOU GET NO CUSTOMERS"
746 PRINT "YOU EARN NO MONEY - CHOOSE ANOTHER JOB"
750 F = 1
755 GOTO 340
765 PRINT "BUSINESS IS BOGGING"
766 PRINT "YOU ONLY EARN $18.00"
770 E = E + 18
775 GOTO 340
780 PRINT "SOME CUSTOMERS ARE UNHAPPY WITH STREAKED
      WINDOWS"
781 PRINT "YOU ONLY CAN CHARGE $1.25 PER CAR"
785 E = E + 20.75
790 GOTO 340
800 REM ** INSTRUCTIONS FOR GAME
805 PRINT "YOU HAVE 6 WEEKS TO EARN $145 FOR A SKI
      WEEKEND"
810 PRINT "YOU HAVE A CHOICE OF 3 JOBS"
815 FOR J = 1 TO 3

```

```

820 GOSUB 400
821 PRINT
825 NEXT J
826 PRINT "INPUT THE JOB NUMBER 1 - 3 TO CHOOSE A JOB"
830 PRINT "IF YOU WORK HARD AND HAVE A LITTLE LUCK"
835 PRINT "YOU WILL HIT THE SLOPES IN 6 WEEKS"
840 PRINT "HAPPY HOLIDAYS"
845 GOTO 130
999 END

```

**Note:** This is a typical computer game. You can write one like this with the programming statements you learned in the earlier chapters.

Lines 100-120: Allows a person familiar with this game to skip the instructions.

Lines 130-145: Initializes (sets the starting values of) the variables used in the game.

Lines 150-210: Gets job choice and branches to that job in program.

Lines 500-599: If player chooses the paper route, the computer randomly determines the money earned.

Lines 600-685: If player chooses baby-sitting, the computer randomly determines the money earned.

Lines 700-790: If player chooses the car wash, the computer randomly determines the money earned.

Lines 800-825: Prints the instructions.

Lines 826-845: Player won, prints congratulations.

### VARIABLE LIST

T—Total amount needed to earn for vacation

W—Week number

E—Money earned so far

J—Job number input

F—Used to control player's options; if  $F=1$  the player has to change jobs

R—Random number determining outcome for each week

*Special credit to Bill Rehor and Slaton Lipscomb*

---

# BACKPACK

---

You are in a backpack race across the Sierras. In your pack, you have 5 days of food and a canteen of water. See if you can be the first hiker at the finish line!

You can add other obstacles to this game if you want to make it really tough.

## SAMPLE RUN

### BACKPACK RACE

#### WANT INSTRUCTIONS?Y

YOU ARE IN A 50 MILE CROSS-COUNTRY BACKPACK RACE  
YOU START WITH 5 DAYS OF FOOD AND 1 DAY OF WATER  
YOU CHOOSE YOUR HIKING SPEED EACH DAY AS FOLLOWS

- 1 - HIKE LESS THAN 10 MILES
- 2 - HIKE 10 MILES
- 3 - HIKE MORE THAN 10 MILES
- 4 - STAY IN CAMP

YOU LOSE AND HAVE TO RADIO FOR HELP IF  
YOU GO WITHOUT WATER FOR 2 DAYS  
OR YOUR FOOD SUPPLY IS LOW FOR 2 DAYS

#### DAY 1

##### CHOOSE YOUR SPEED?3

HIKING AT THE RATE OF 11 MILES/DAY

YOU TAKE THE WRONG TRAIL - LOSE 2 MILES

YOU MEET OTHER HIKERS - THEY GIVE YOU FOOD AND WATER

AFTER A HARD DAY, YOU MAKE CAMP

YOU MADE 9 MILES TODAY

YOU HIKED 9MILES TOTAL

YOU SLEPT WELL - READY FOR A GOOD DAY OF HIKING

#### DAY 2

##### CHOOSE YOUR SPEED?2

HIKING AT THE RATE OF 10 MILES/DAY

YOU TAKE THE WRONG TRAIL - LOSE 2 MILES

YOU FIND A SHORTCUT - GAIN 2 MILES

AFTER A HARD DAY, YOU MAKE CAMP

YOU MADE 10 MILES TODAY

YOU HIKED 19MILES TOTAL

YOU SLEPT WELL - READY FOR A GOOD DAY OF HIKING

#### DAY 3

##### CHOOSE YOUR SPEED?3

HIKING AT THE RATE OF 14 MILES/DAY

YOU FIND A SHORTCUT - GAIN 2 MILES

YOU CROSS A STREAM AND FILL YOUR CANTEEN

AFTER A HARD DAY, YOU MAKE CAMP  
 YOU MADE 16 MILES TODAY  
 YOU HIKE 35 MILES TOTAL  
 YOU SLEPT WELL - READY FOR A GOOD DAY OF HIKING  
 DAY 4  
 CHOOSE YOUR SPEED?2  
 HIKING AT THE RATE OF 10 MILES/DAY  
 YOU FIND A SHORTCUT - GAIN 2 MILES  
 YOU TAKE THE WRONG TRAIL - LOSE 2 MILES  
 AFTER A HARD DAY, YOU MAKE CAMP  
 YOU MADE 10 MILES TODAY  
 YOU HIKE 45 MILES TOTAL  
 BEAR CARRIES OFF YOUR PACK - LEAVES YOUR CANTEEN  
 DAY 5  
 CHOOSE YOUR SPEED?3  
 HIKING AT THE RATE OF 12 MILES/DAY  
 YOU CROSS A STREAM AND FILL YOUR CANTEEN  
 YOU TAKE THE WRONG TRAIL - LOSE 2 MILES  
 AFTER A HARD DAY, YOU MAKE CAMP  
 YOU MADE 10 MILES TODAY  
 YOU HIKE 55 MILES TOTAL  
 CONGRATULATIONS!! YOU REACHED THE FINISH LINE!!  
 YOU WON - FIRST PLACE!!  
 WANT TO TRY AGAIN?N

## PROGRAM LISTING

```

95 DIM A$(1)
100 PRINT "BACKPACK RACE"
105 PRINT "WANT INSTRUCTIONS";
110 INPUT A$
115 IF A$ = "Y" THEN 650
120 F = 5
122 G = 0
125 A = 1
130 W = 1
135 D = 5
140 M = 0
145 S = 0
150 E = 0
155 PRINT "DAY ";A
160 A = A + 1
185 PRINT "CHOOSE YOUR SPEED";
190 INPUT J
195 ON J GOTO 200,215,230,300
200 LET S = INT ( RND (1) * 8 + 1)
205 PRINT "HIKING AT THE RATE OF ";S;"MILES/DAY"
210 GOTO 350
215 LET S = 10
  
```

```

220 PRINT "HIKING AT THE RATE OF ";S;" MILES/DAY"
225 GOTO 350
230 LET S = INT ( RND (1) * 15 + 1)
235 IF S < 11 THEN 230
240 PRINT "HIKING AT THE RATE OF ";S;" MILES/DAY"
245 LET R = INT ( RND (1) * 6 + 1)
250 ON R GOTO 255,270,350,350,350,350
255 PRINT "GET BLISTERS - ONLY MAKE 4 MILES"
260 S = 4
265 GOTO 470
270 PRINT "SLIP ON A LOG - LOSE PACK IN STREAM"
275 F = 0
280 GOTO 470
300 LET S = 0
305 PRINT "STAYING IN CAMP TODAY"
310 LET R = INT ( RND (1) * 3 + 1)
315 ON R GOTO 320,330,485
320 PRINT "RANGERS ARRIVE AND CARRY YOU OUT"
325 GOTO 765
330 PRINT "YOU RECUPERATE QUICKER THAN YOU THOUGHT"
335 PRINT "YOU CAN HIKE 4 MILES TODAY"
340 S = 4
345 GOTO 350
350 REM **WHAT HAPPENS TO YOU DURING YOUR DAY
355 FOR I = 1 TO 2
360 LET R = INT ( RND (1) * 6 + 1)
365 ON R GOTO 370,385,400,415,430,450
370 PRINT "YOU TAKE THE WRONG TRAIL - LOSE 2 MILES "
375 S = S - 2
380 GOTO 465
385 PRINT "YOU FIND A SHORTCUT - GAIN 2 MILES"
390 S = S + 2
395 GOTO 465
400 PRINT "THE TRAIL IS WASHED OUT - LOSE 4 MILES"
405 S = S - 4
410 GOTO 465
415 PRINT "YOU CROSS A STREAM AND FILL YOUR CANTEEN"
420 W = 1
421 E = 0
425 GOTO 465
430 PRINT "YOU SLIP OFF THE TRAIL - HURT YOUR LEG"
435 PRINT "YOU CAN ONLY HIKE 4 MILES"
440 S = 4
445 GOTO 485
450 PRINT "YOU MEET OTHER HIKERS - THEY GIVE YOU FOOD
AND WATER"
455 F = 5
460 W = 1
461 E = 0

```

```

462 G = 0
465 NEXT I
470 PRINT "AFTER A HARD DAY, YOU MAKE CAMP"
475 IF S > 0 THEN 485
480 S = 0
485 M = M + S
490 PRINT "YOU MADE ";S;" MILES TODAY"
495 PRINT "YOU HIKED ";M;" MILES TOTAL"
500 IF M > = 50 THEN 720
505 D = D - 1
510 F = F - 1
515 IF F + 1 > = D THEN 540
521 IF G > 0 THEN 525
522 G = 1
523 GOTO 535
525 PRINT "YOU RAN OUT OF FOOD"
530 GOTO 705
535 PRINT "YOU ARE RUNNING LOW ON FOOD - BETTER SPEED
UP TOMORROW"
540 IF S > 6 THEN 585
545 PRINT "YOU HIKED LESS THAN 7 MILES TODAY"
550 PRINT "YOU MUST MAKE A DRY CAMP AND DRINK FROM YOUR
CANTEEN"
555 IF W > 0 THEN 575
565 PRINT "WHOOOPS - YOU DON'T HAVE ANY WATER IN YOUR
CANTEEN!!"
567 IF E = 1 THEN 705
570 PRINT "YOU HAVE TO GET WATER TOMORROW OR YOU LOSE"
575 E = 1
580 W = 0
581 GOTO 585
582 W = 1
583 E = 0
585 LET R = INT ( RND (1) * 6 + 1)
590 ON R GOTO 595,610,640,640,640,640
595 PRINT "BEAR CARRIES OFF YOUR PACK - LEAVES YOUR
CANTEEN"
600 F = 0
605 GOTO 155
610 PRINT "SQUIRRELS GET INTO YOUR PACK - EAT 2 DAYS
OF FOOD"
615 F = F - 2
620 GOTO 155
640 PRINT "YOU SLEPT WELL - READY FOR A GOOD DAY OF
HIKING"
645 GOTO 155
650 PRINT "YOU ARE IN A 50 MILE CROSS-COUNTRY BACKPACK
RACE

```

```

655 PRINT "YOU START WITH 5 DAYS OF FOOD AND 1 DAY OF
WATER"
660 PRINT "YOU CHOOSE YOUR HIKING SPEED EACH DAY AS
FOLLOWS"
665 PRINT "    1 - HIKE LESS THAN 10 MILES"
670 PRINT "    2 - HIKE 10 MILES"
675 PRINT "    3 - HIKE MORE THAN 10 MILES"
680 PRINT "    4 - STAY IN CAMP"
690 PRINT "YOU LOSE AND HAVE TO RADIO FOR HELP IF"
695 PRINT "YOU GO WITHOUT WATER FOR 2 DAYS"
696 PRINT "OR YOUR FOOD SUPPLY IS LOW FOR 2 DAYS"
700 GOTO 120
705 REM **HAVE TO DROP OUT OF RACE
710 PRINT "RADIO FOR HELP, YOU AREN'T GOING TO MAKE IT"
715 GOTO 765
720 REM **YOU REACH THE FINISH LINE
725 PRINT "CONGRATULATIONS!! YOU REACHED THE FINISH
LINE!!"
730 R = INT ( RND (1) * 3)
735 R = R * 10
740 IF R > 0 THEN 755
745 PRINT "YOU WON - FIRST PLACE!!"
750 GOTO 765
755 PRINT "THERE ARE ";R;" HIKERS AHEAD OF YOU"
760 PRINT "BETTER LUCK NEXT TIME"
765 PRINT "WANT TO TRY AGAIN";
770 INPUT A$
775 IF A$ = "Y" THEN 100
780 END

```

**Note:** Notice that different things can happen depending on how fast the hiker chooses to travel.

### VARIABLE LIST

- F —Days of food left — starts with 5
- W —Water in canteen — starts with 1
- D —Days left to hike
- M —Total miles traveled
- S —Miles traveled during the current day
- T —Flag used to control options, if  $T > 0$ , program specifies speed
- J —Hiking speed input
- E —Set to 1 if player runs out of water. If player is out of water for 2 days in a row, forces player to radio for help.
- G —Set to 1 if player runs out of food. Forces player to radio for help if player is out of food for 2 days in a row.

---

# APPENDIX A

---

## Differences Between the Versions of BASIC

**BREAK KEY** Every computer has a break—or program interrupt—key. However, some use a different name, or require a combination of keys. Use the programmer's reference manual included with your computer to determine what it uses. In this text, we have adopted the convention of referring to this key as the break key.

**RANDOM NUMBER FUNCTION (RND)** The format for this function varies. The Apple, HP-2000, Atari, Osborn and IBM use RND(1), the TRS-80 uses RND(0), and the TI uses RND. The Osborn and IBM computers also require the statement RANDOMIZE in the beginning of the program in order to get a different series of random numbers each time the program is run.

**PRINT statement using , (comma) or ; (semicolon)** The spacing produced by the , and the ; varies with each version of BASIC. Therefore, some of the programs, such as MVP in Chapter 10, which print a heading and columns of numbers, may not line up on your computer as shown in the sample output. Experiment with your own computer to determine the proper spacing.

**IBM BASICA, Osborn CBASIC, BASIC-Plus and Altair Extended Basic** These versions of BASIC allow string arrays. The programs in this text use standard BASIC which does not have string arrays. In order to run these programs, leave out the string variables in the DIM statements.

**Atari-BASIC** Does not have a TAB function. You can convert programs that use this function as follows:

a) replace TAB(10); "\*"

-with-

PRINT "(10 spaces)"; "\*"

b) replace TAB(W); "\*"

-with-

FOR I = 1 TO W-1

PRINT " ";

NEXT I

PRINT "\*"

**TI-BASIC** Does not allow use of DIM statement with string variables. In order to run the programs in this text, simply eliminate the string variables in the DIM statement in each program.

—Does not have string functions needed in order to combine strings. Therefore, the following programs in Chapter 5 cannot be run on the TI:

RHYME

AUTHOR

**Radio Shack—TRS-80 Color BASIC** Does not use the word LET in assignment statements. Therefore, leave this word out when entering these statements. For example:

Instead of: LET A = B + 1

Use: A = B + 1

**HP-2000 BASIC** The format for the multiple GOTO statement is different from the one used in the programs in this text.

Instead of: ON X GOTO line #, line #, ...etc

Use: GOTO X OF line #, line #, ...etc





# ***MORE BASIC THAN ABC***

KIDS! Write your own computer games!

PARENTS! Your kids can show you how!

TEACHERS! Use this book as a text for programming classes!

There's so much you can do on a computer—once you know how. This easy-to-follow guide can serve as an introduction to any computer which uses the BASIC programming language. BASIC FUN begins with the most simple programs for beginners and progresses to more difficult programs for expert programmers. There's everything from drawing a rocket ship that blasts off your video terminal to figuring out time differences around the world, and calculating your favorite ballplayer's batting average to teaching your computer a magic trick. These programs are for anytime, anywhere—home *and* school—and anybody who's ever been interested, curious, or fascinated by computers.

AVON  CAMELOT

PUBLISHED BY AVON BOOKS



ISBN 0 7100 10029 5