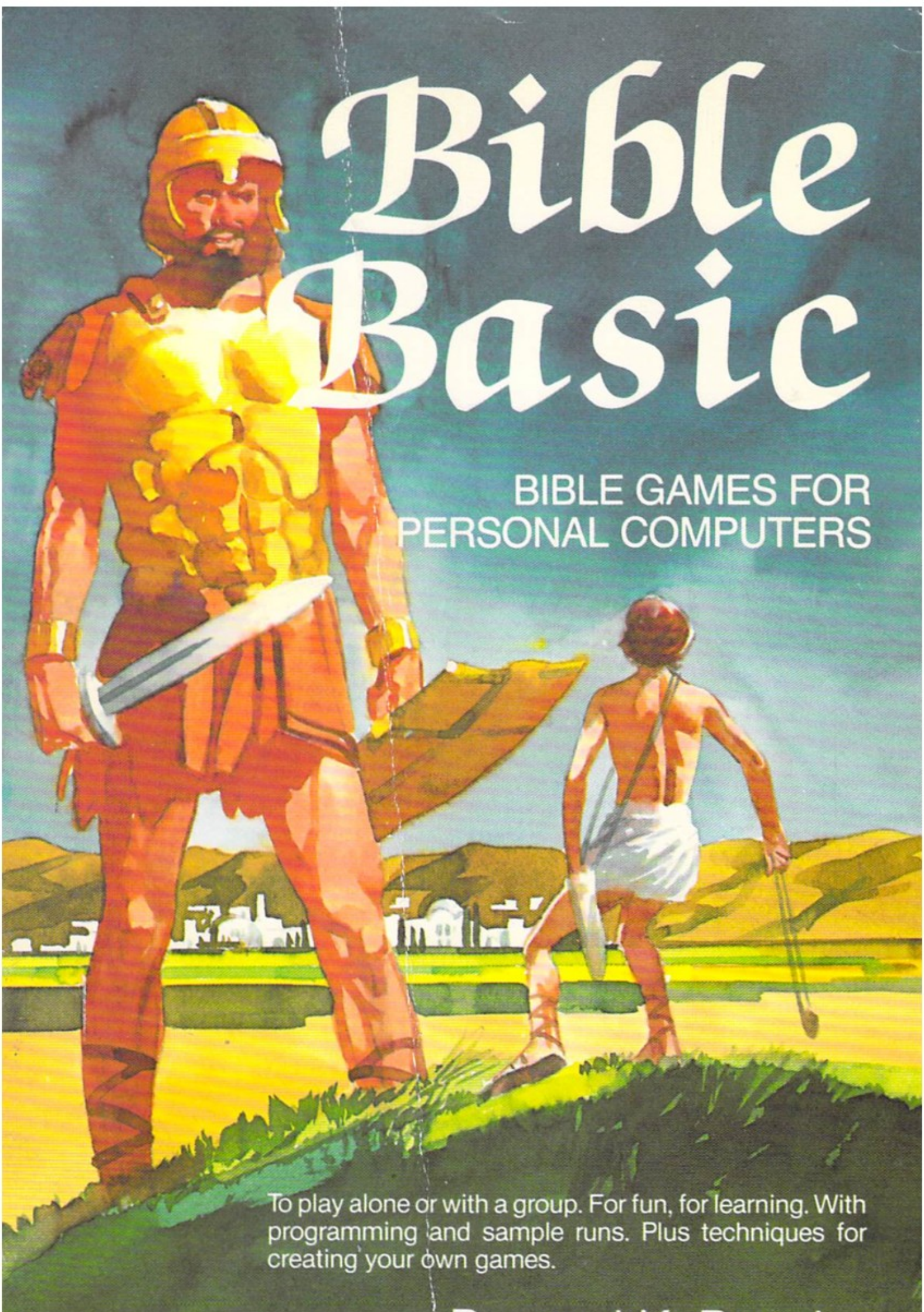
Bible Basic

BIBLE GAMES FOR PERSONAL COMPUTERS



To play alone or with a group. For fun, for learning. With programming and sample runs. Plus techniques for creating your own games.

Bible Basic

BIBLE GAMES FOR
PERSONAL COMPUTERS

Bible Basic

BIBLE GAMES FOR PERSONAL COMPUTERS

Bernard K. Bangley

Harper & Row, Publishers, San Francisco

The Bible-related games in this book can be adapted to any personal computer that uses a form of BASIC. Reference is made here to several of the most popular models of personal computers, including the VIC-20™, Commodore 64™, TRS-80®, Apple II™, and Atari 400/800™. Apple® is a registered trademark of Apple Computer, Inc. ATARI® is a registered trademark of Atari, Inc. TRS-80® is a registered trademark of Tandy Corporation. VIC-20™ and Commodore 64™ are trademarks of Commodore Electronics Limited.

Designed by Michael Rogondino
Illustrations by Dick Cole

Library of Congress Cataloging in Publication Data
Bangley, Bernard, date
BIBLE BASIC.
1. Bible games and puzzles. 2. Computer games. 3. Basic (Computer program language). I. Title.
GV1507.B5B36 1984 220'.07 83-48461
ISBN 0-06-250042-2

CONTENTS

FOREWORD

By Charlie Shedd

"The stream of history is passing through my consciousness to farther reaches."

Words of an unknown sage, long gone. But isn't his quaint phrase a good testing-place for any of us in any generation?

I recently heard a debate on computers. It was nothing more than one of those living room arguments with no conclusion. The evening was closing down and we were waiting for dessert.

Somehow the conversation turned to education and one group (headed by an arch right-winger) contended, "Do you know why today's kids can't read, can't write, can't think? It's computers, that's why. Et cetera, blah, blah, blah."

There were plenty to take the other side. One highly successful executive saw it from an altogether different angle. "Computers," she said, "aren't bad in themselves. When we use them right, they really can do a lot of our thinking for us. But what's the matter with that? If they free our minds for more important things, isn't that a real plus? When education learns how to handle computers, sees all their possibilities, they can even lead us to whole new ranges of the human mind."

Say it again, old friend: "The stream of history is passing through my consciousness to farther reaches."

So here comes Bernard Bangley with Bible games for computers. The Lord tells us often in his Book that he can make his presence known to us in *all* things. That being the case, I wonder if the executive isn't straight on target. Rightly used, computers can enable us to have more time for more important things. And what is more important than family togetherness, family "think" times, family fun?

I once wrote a column for *Teen* magazine. It was a question-and-answer piece with thousands of letters coming in. From my reading of that voluminous input, some things came ultra clear—one

of which was that the stable young are no accident. Almost without exception those of strong character come from homes with warm recall. There may have been tragedy, struggle, divorce, and more struggle, but alongside all such negatives were some very real positives. Such as? Such as "Somebody cared about me. Somebody cared enough to spend time with me, to share, to have fun."

Many of us believe that the Bible is history's all-time word of truth. Here are the ultimate answers for men, women, children, you, me, people as they should be, civilization as God meant it.

If, then, it is possible to combine happy recall with directives out of the Book, shouldn't we all celebrate? And what better way to do it than by using today's tools for a better tomorrow?

Imagine what can happen in the minds of our sons and daughters when we spend computer game-time together on any one of these: "People Who Met Christ," "Bible Quiz Code," "Which Book?" "Memory Verses," "Famous Bible Women," or "The Perils of Paul."

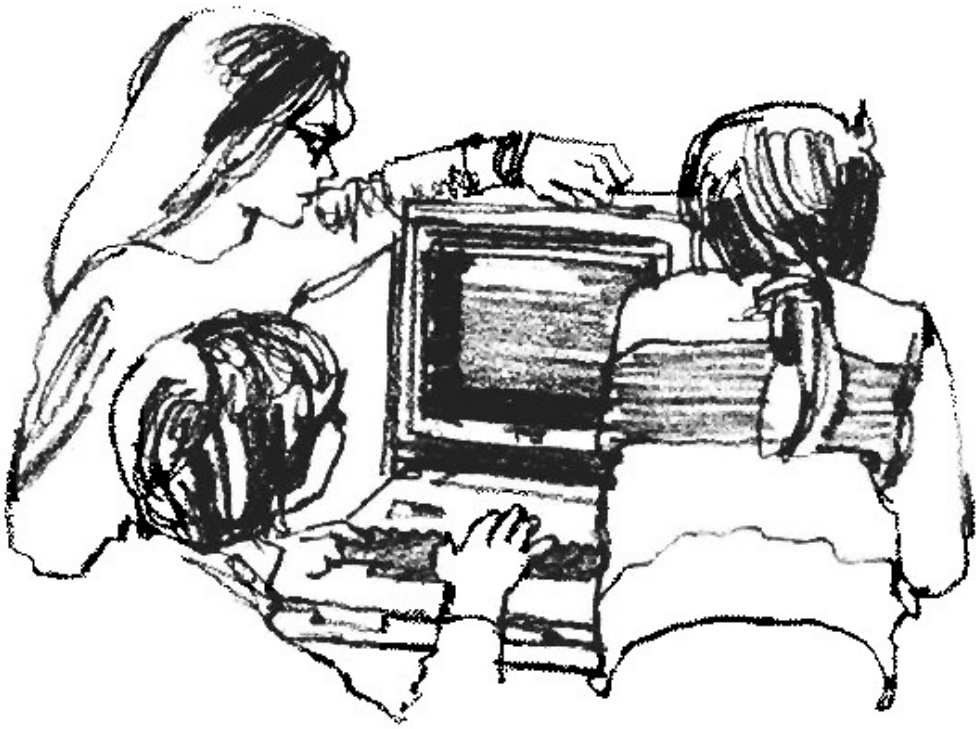
I love this book. I hope it has a wide distribution. Bringing God's word into the computer scene could be a significant contribution to many lives—especially if we use it to "pass the stream of history through our consciousness to farther reaches."

ACKNOWLEDGMENT

The programs in this book owe their precision engineering to my seventeen-year-old son, David. My imagination was able to conceive the games, but in many cases I ran into impossible bugs in my attempts to make them work. David was able (sometimes in an instant, sometimes in a few days) to discover my mistakes and correct them. "People Who Met Christ," "Where Are We Now?" "Twenty Questions," "Things in the Bible," "Memory Verses," "Captive!" "Seven Churches," and "Bible Pairs" are entirely his original creations. All I did was add the DATA where necessary.

His crowning achievement is "Captive!" I told him what I wanted, and then stood back and watched him develop it over a three-month period. It was a difficult challenge. He won.

Bernard K. Bangley



1

COMPUTERS CAN BE FUN

Personal computers are the most astonishing introduction into our lives since television. Modern technology has made available to ordinary people equipment that would have cost many thousands of dollars a few years ago. ENIAC's 18,000 vacuum tubes were anything but compact, and its memory and functions could not begin to compete with a modern microprocessor. Low-cost high-technology computers have become a reality in our day.

The market is booming. Thousands upon thousands of personal computers are being bought each month. As the quality continues to rise and the prices keep falling, home computers are becoming

When we first heard about personal computers in the home, we were told in the publicity releases that we would be able to keep an inventory of the items on our pantry shelves, order macaroni, pay bills, dial the telephone, and control the furnace thermostat with them. No wonder they didn't catch on! Who needed one?

The truth is that the few personal computers taken home were not being used for such mundane tasks at all. They were being used for leisure-time recreation. The hobbyist could invent and play some fascinating games.

While this quiet revolution was taking place among a few highly devoted computer buffs, the advent of video games produced a new arcade industry. Space Invaders™, Pac-Man™, and Asteroids™ swept the country. An accurate figure is hard to come by, but it is acknowledged that young people are dropping quarters in excess of a billion dollars each year. Video games have now become a part of our culture. Home television sets are being watched more and more, not for the commercial networks' latest situation comedy, but for the video game being fed it by a cartridge plugged into a home computer.

The word is out: personal computers can be fun. In fact, they can be so much fun that it is possible to lose oneself in the creation and execution of a program so that hours pass like minutes. No one can say exactly where all of this will lead and what will become of us if this becomes some kind of national addiction. These are heavy questions that must be pondered elsewhere.

It occurred to me that, although some churches were beginning to use computer technology for record keeping and standard business applications, nobody had yet seen the magnificent opportunity these machines give us for teaching and recreation in the Christian context. If one could program them to defend against aliens from outer space, one could also program them to fight the Battle of Jericho. Moreover, if children were naturally attracted to computers, why not let them absorb a little biblical literacy at the same time?

There is a saying that it's not work if it's fun. Fortunately, the archaic methods of teaching by rote and hand-spanking are not as prevalent as they used to be. Unfortunately, those stern pedagogical methods have been replaced in our day with threats of failure, criticism, and even ridicule. This is still teaching by punishment. Those of us who teach really need to heed the students of educational psychology. Any kind of reinforcement, an event that follows a behavior that makes the person want to do it again, is a teaching tool superior to punishment.

As I worked to develop the Bible games included in this book, I saw how a person of any age could begin learning things about the Bible, almost unconsciously, while having a good time. Fixing the order of the sixty-six books of the Bible in one's memory can be a difficult chore if all it gets you is a gold star on the bulletin board. But if familiarity with that order, and the proper spelling of each title, allows you to match wits with a talkative and enthusiastic computer, it becomes a matter of genuine pleasure. The player wants to recall important biblical facts in order to outwit the machine! This is not a hypothesis; this is the way it actually turns out in the real world. Christian educators, take note. The problem is in getting one player to yield the computer to another person.

The Bible games for computer in this book are written on a level that allows even a beginning programmer to enjoy immediate success. As will be pointed out in the next chapter, anyone with advanced experience will be able to take these rather bare programs and turn them into works of art.

These Bible games have a place in any Christian home. A family can enjoy hours of quiet recreation together with any one of them. As the weeks go by, a fresh new familiarity with the factual aspects of the games will surprise them all.

The games can also be played to good advantage at church. I have seen them captivate people of all ages. They have caught the attention of senior citizens, middle-aged professionals, and youth groups. Use of such games is still a largely unexplored approach to Christian education, and this approach is crying out for careful study in our seminaries.

Today's younger generation is already familiar with computers, and they are intimate with video games. Having an opportunity to compete with each other in such disarming terms will open vast new approaches to teaching and recreation at church. The games in this book have actually been tested in group situations. A summary of these encounters is included in the introductory text that accompanies the programs. In every case, the response of users of these programs was immediate, positive, and enduring.

One group that experimented with these games was a mixed class of fifteen eighth graders. They met with me for twenty Wednesday afternoons in preparation for making their profession of faith in Jesus Christ and becoming members of the church. One day when they entered the room, I had set up a simple program that tested them on the previous week's subject. That class had necessarily been

almost straight lecture, an attempt to impart a few elemental facts that any responsible church member needs to know.

When the class was seated, I typed RUN, and the computer came alive. It greeted the class, said a few complimentary things, and then asked them what the name *Presbyterian* meant. All of the youngsters immediately became vocal, calling out the answer to the glowing screen. "Governed by elders!" "Ruled by elders!" "Elders!" "Greek for elder!"

Because I knew they would get it right, a simple time-delay loop let the program move on to thank them for their wisdom in appreciative terms. It seemed as though it had responded to their verbal statements. Then it said, "OK, now tell me, who was John Calvin?" Again, the well-informed verbal responses. After another short delay the screen cleared and the computer asked, "And where did he work?" A pause, and then, "That's right! Geneva, Switzerland." The pauses were such that I could enter the class discussion and still make it appear as though the computer were in charge. I knew that if I needed more time for clarification and hint dropping, I could push a button and stop the RUN. The simple program, which I had typed in during the twenty minutes before the class, had led us through a detailed review of some important material, and it had been fun.

And then, as a reward for their patience in dealing with the heavyweight material the week before, I loaded "David and Goliath" from a cassette and let individuals take turns playing it. They called the moves they wanted from their seats, and I entered them at the keyboard. After a few games, serious competition developed and the interest of the class never flagged.

Who can define the limits of such computerized Bible games? There are many creative opportunities. Aside from their valuable function in private and classroom settings, they could also do well with public exposure in unlikely places. Perhaps they could be set up in office waiting rooms or placed in the lobby before church dinners. If people see the games being played, they will want to try a turn for themselves.

Of this I am sure: there will be enthusiasm and laughter. Computers can be fun—even in church.

2

ADAPTING THE PROGRAMS

Each one of the Bible games included in this volume is a direct printout of a debugged, working model in the BASIC language. It is complete and ready to play. If you enter it accurately into your computer and make any necessary minor adjustments as outlined in this chapter, it will certainly work. Each program has been carefully written to avoid those areas of incompatibility among the BASIC languages employed by various computer manufacturers. For instance, there are no POKES, PEEKS, or INKEY$ statements. The programs are intentionally open and lean. This gives you a strong

design. Several of the games also include an expanded version for a particular computer as an example of what can be done. This chapter will give you many hints on how to go about improving the programs given in this book.

Notice first of all that most programs actually begin with line 100. A few elemental instructions are suggested in the opening lines, such as PRINT and REM statements, but in reality the first ninety-nine lines are a clean slate on which you can create something beautiful of your own. Young people, especially, respond to a jazzy opening. Animate some graphics, write some music, or use a variety of electronic blips that will help bring the Bible into the twentieth century. Using time loops, you can let the instructions appear on the screen one line at a time. This will be far easier to read than a screen suddenly loaded with verbiage. You can scroll the instructions by from bottom to top. At the very least, be sure to center the titles and other opening language in an attractive manner. Drop down a few lines from the top of the screen and add clean spacing to center words. Use a few asterisks and underscores to make some of the important features stand out. If your computer has color capability, use it to good advantage. The idea is to catch the player's attention from the first moment the program starts to run.

Incidentally, after the opening instructions have been seen and absorbed, they can become boring with repeated exposure. This is true no matter how clever you make them. Once a person is actively involved in the game and chooses to try it again for a higher score, it is frustrating to have to sit through a show intended for beginners. For this reason I suggest that most programs have a GOTO 100 (or whatever) line if the final INPUT asks for another turn. Of course, it is a simple matter to add a third character string to the final INPUT that gives the player an option of seeing the instructions again. You can see an example of this in "Famous Bible Women," beginning at line 1010.

If the program begins with a DIMension statement, be sure to put it *before* any instructions that may be skipped. A second encounter with a DIM statement will confuse and upset your computer. The only exception is in a case like "Twenty Questions," in which a player's request for another game results in a RUN command.

Throughout the programs you will discover numerous opportunities for audio and graphics. It is assumed that you are familiar with your own computer and will know how to introduce these features. For instance, an electronic "beep" can be added to each pressing

ing sound, your instruction manual will certainly tell you how to POKE it into fewer than the ten lines left open between major code lines.

If you want to try drawing characters on the screen, such cartoons can add a lot of action and enjoyment to the playing of a game. Try creating an upright and menacing Goliath, and then draw him prone and defeated. Let the beaten Goliath appear on the screen in conjunction with the message, "Goliath is dead!"

Another extremely useful extra touch is to make warnings and failures stand out boldly so that the words are not lost on the screen among other lines. You can begin these notices with asterisks or enclose them in printed parentheses. The very best effect is to reverse the contrast for these lines only. In APPLE II you begin the line with the command INVERSE and finish it with NORMAL. With Commodore products, you press the CONTROL key (once you are inside quotation marks) and then the REVERSE ON key, which will print the light *R* in a dark box as a reminder of what you have done. Then, after you have typed the line, press CONTROL and REVERSE OFF. Enclose the light dash in the dark box inside the final quotation mark and the entire line will be surrounded by a dark border when the program is run. You can see a specimen of this in the VIC-20 expanded version of "David and Goliath." Such touches make these exciting lines jump out at the player. The reward is far greater than the effort.

Speaking of rewards, some sort of pat on the back is an essential element for an enjoyable Bible computer game. A high score is just a start. You can add degrees of rank.

LUCKY
BEGINNER
NOT BAD
GOOD
QUITE GOOD
OUTSTANDING
SCHOLAR
WIZARD

Rank can be assigned on the basis of score spread using IF THEN statements with greater or lesser parameters. Notice how this is done in "Memory Verses." If you add a little music to the higher

repeatedly and well. If your computer has the memory capacity for it, you can even reward the player with a choice of prizes by asking the player to press any key. Some examples:

```
IF P$="S" THEN PRINT "STARS IN YOUR CROWN!"
IF P$="W" THEN PRINT "WATER FROM A ROCK"
IF P$="H" THEN PRINT "HOUSE OF THE LORD FOR-
  EVER"
```

A complete specimen of this is included in the Appendix, which uses "prizes" that are humorous in the state of Florida, where these games were developed.

One excellent way to reward the player is by tacking on a little subroutine with an enjoyable tic-tac-toe game. On some computers you could even turn the keyboard into a piano.

Computer games provide some of the best imaginable opportunities for positive reinforcement. Let a high scorer write identifying initials on the screen. This can be especially sweet if, in the process, another name can be bumped lower.

When you improve these programs, be careful to give the best reinforcement for the *correct* answers. Some programmers have made the mistake of putting the most attractive rewards for the wrong answers. You can be sure that the player, once this fact is discovered, will enter wrong answers just to watch the show!

To be most effective, any prize must come immediately after the right answer or series of answers. Psychologists have discovered that an interval of half a second is ideal. Moreover, the reward itself must not become a bore. Make it spectacular, if you can, but keep it short and sweet—*unless* you are going to invite the player to participate.

The differences in the various BASIC languages now on the market have forced a certain style upon the games as printed. It is not necessary for you to maintain this universal style in order for the programs to work for you. In fact, you are encouraged to break out of it at every opportunity that occurs to you.

Some of these games will scroll by, teletype fashion, on each player INPUT. One advantage of this form of presentation is that it lets you compare your present move with your previous one. But in many cases it will be far better to clear the screen each time around. Unfortunately, each manufacturer has devised a different method to

you will use CLS. On a VIC-20 or a Commodore 64 you will make a PRINT statement, add a quotation mark, press the shift key, and then put in the little heart in a dark box by touching the CLEAR/HOME key. Using APPLE II, you will enter the word *HOME* or use CALL-936. Such commands will be thoroughly familiar to anyone who has been playing with a personal computer for only an hour. The programs in this book have eliminated all such confusing clutter by printing the COMMODORE PET symbol for clearing the screen where it is essential to do so. A REM statement follows a colon (:), instructing you to replace the entire line with your version of the screen-clearing code.

Another example of the shortcuts that have been neglected for the sake of universality is the matter of spacing between lines. PRINT:PRINT:PRINT will result in three blank lines on anyone's computer. If you are working in PET BASIC it will be far simpler to add three cursor down marks inside the first quotation mark after a PRINT statement. These show up as a reverse *Q* in the programs adapted for the VIC-20. With other brands of computer you can do beautiful alignment and spacing with TAB functions and PRINT AT statements.

Great care has been given to including in this book programs that use only commonly shared BASIC language. Beyond the usual mistakes, like entering a semicolon where you meant to put a colon, you should have no difficulty debugging any of the games. If any problems develop when you try to RUN a program that you have elaborated, pay close attention to the IF THEN statements. There may be a peculiar requirement regarding the format of these statements for your personal computer. In rare instances you may need to add a GOTO statement. There is also a possibility that your computer may want a special code for the selection of a random integer. A typical form in this book is N=INT(RND(1)*20)+1. The necessity for such a formula is so widespread that there is little possibility that you have not used it or something similar even if you are a complete neophyte. If the program will not run as printed, change the RND code to resemble something you've used before. Just be sure to keep the big number the same as that in the published program.

The chances are very remote that your computer will require an actual spelling out of a LET statement. If you normally use them, then change any line like W=1 to LET W=1. Read your instruction manual! If you understand exactly what your computer is looking

Most personal computers will automatically add a question mark after an INPUT statement. If yours does not, add them where they seem appropriate.

Time loops work at different speeds on the various brands of computers. Experiment. If you don't like the results of FOR T = 1 TO 500: NEXT T, try enlarging or reducing the greater number.

When you read the introductory material to each game, you will find some specific suggestions for adding flesh to the skeleton. Take these suggestions seriously. You can add a lot of sport to the player's options with a little judicious doctoring. If you make a mess of it, it is a simple matter to remove any additional lines that got you into trouble. If you succeed, you will have the satisfaction of being a cocreator. Several examples of games that have been enhanced for use on a VIC-20 and a TRS-80 are included for illustrative purposes. If you happen to own one of these popular computers, by all means enter that version for your own use.

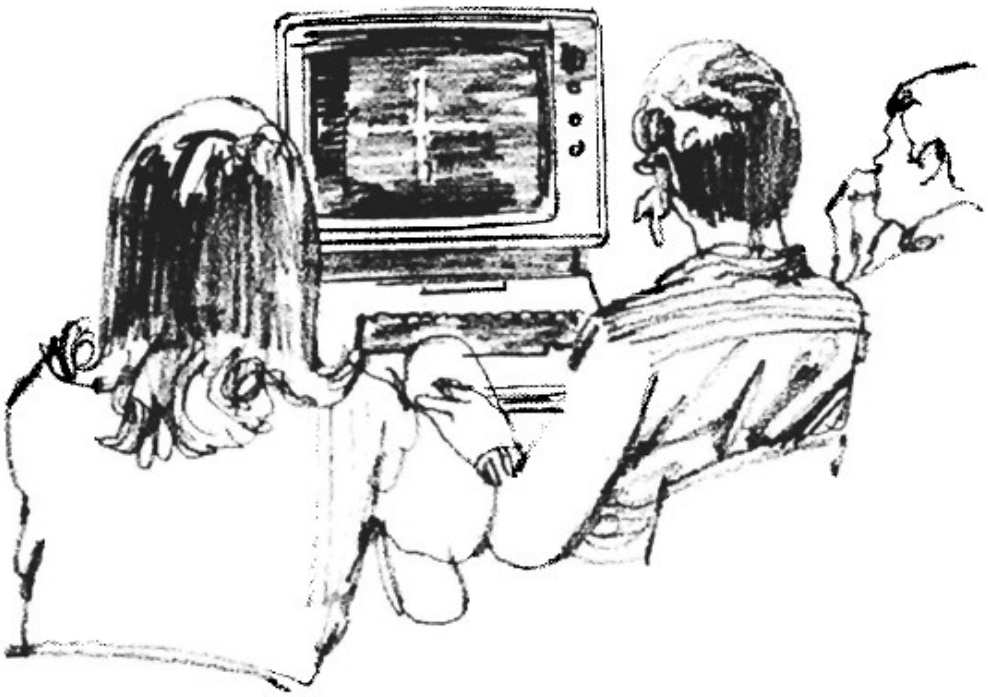
One final suggestion, please. Computers are, by nature, cold and impersonal. An officer in my congregation described a telephone sales call she received that had been dialed by a computer and spoken by a voice synthesizer. Her response was a mixture of annoyance and terror. Our country has already witnessed a sort of computer rebellion in the marketplace, with customers doing everything in their power to get some kind of personal attention. The last thing we need to do is to add to this depersonalization.

One beautiful way to avoid this is by the addition of an opening INPUT statement to many of the games. Say something friendly like, "Hello, friend. What is your name?" Then let the player INPUT a character string that can be brought back into play at intervals throughout the game. By asking the computer to PRINT N$, or whatever you have decided to call it, you can then let it say things such as "Mary Jones is a genius!" or "Not bad, Mary Jones, but guess again." Not only will this add a little warmth and humanity to the game, it has the added feature of letting the player vent some frustration in a harmless way. If Mary Jones wants to type in "BUBBA" as her name, the stupid computer will accept it as fact and go on making a laughingstock of itself by recalling the error in all innocence throughout the game. Once young people have discovered this, you can be sure they will refrain from using their real names ever again! Examples of this technique can be seen in the games, "Which Book?" and "Forty-two Kings."

Use your imagination and your intimate knowledge of your

personal computer to dress up each one of these stock programs. The possibilities are limitless; the results can be magical.

Now GOTO the games themselves. Enjoy loading them into your own computer's memory. Once you have satisfied yourself that you have tailored and refined them as well as you possibly can, SAVE them on tape or disc. They will provide you and your friends with hours of fun, and you will be learning more and more about the Bible in one of the most delightful ways ever conceived.



3

STARTING AT CALVARY

Even if you are already a computer whiz kid, spend some time with this chapter. An extremely brief four-line program will be used to demonstrate some of the aspects of customizing the games in this book. These four lines will also be useful to you later as an impressive subroutine at any point of the program of your choice.

Turn on your computer and enter the following:

```
10 FOR X=1 TO 19
20 IF X=6 THEN PRINT TAB(6); "**********":GOTO 40
```

Now RUN it. You will see a cross quickly drawn upon your screen, six spaces from the left margin.

Before you can expect to customize this symbol, you must first thoroughly understand the program that produced it. LIST the program and examine it as you read. Line 10 sets up a loop that cycles back through line 40. The fact that X=1 TO 19 means that it will go around the loop nineteen times before it stops.

The first and last lines by themselves would produce no visible results, even though the computer would obey your command to RUN. Something must be put in a PRINT statement in order for you to see anything on the screen. Line 30 tells the computer to print an asterisk on each of the nineteen lines of the loop, ten spaces from the left margin. Increase the number in parentheses after TAB, and this vertical line of asterisks will move to the right one space for each number. Decrease it, and it will move to the left.

Line 20 prints the horizontal arm of the cross. It will produce exactly as many asterisks as you have enclosed in the quotation marks. In the printed program there are nine asterisks, which allows for one in the middle and four on each side. Therefore, to get this line centered with your vertical line, the TAB number must be exactly four less than the TAB number in line 30. In the present case, line 20 reads six and line 30 reads ten, a difference of four.

Now for your first experiment! Move the cross either to the right or the left by changing the two TAB numbers in lines 20 and 30. Use any number you like, up to the limit of your screen, but be sure to keep the number in line 20 exactly four smaller than the number in line 30. Once you have moved it around a bit, try centering it on your screen. Remember the numbers you use. Write them into the program printed above. They may be very useful to you in the future.

The next alteration you can try is the character used to draw the cross. Asterisks are fine, but sometimes they are overworked. This particular symbol is very effective when made with the letter *O*. Replace the asterisks in lines 20 and 30, one for one, with the letter *O* and watch what happens when you RUN it.

Don't stop with that! Go ahead and try some other keyboard characters. Plus marks are good. Try the letter *X*. Then use your imagination. Find out which characters look best to you and make a note of them for future reference.

Now you can really start getting fancy! Try enlarging the cross. Remember, if you expand it one way, you will also need to expand

20 an *odd* number. Otherwise, the horizontal arm will not appear centered on the vertical. There are nine characters in the printed program. Try eleven, thirteen, fifteen, or any number you like. Now increase the length of the loop in line 10. Nineteen will not be enough. Increase the final number in proportion to the increase in line 20. This will stretch the vertical bar one line for each additional numerical increment. Bear in mind that there is a limit to the vertical dimension of your screen.

Take a look at the results of your experiment. You will notice that awkward things will begin to happen to the proportions of the cross as you make it larger. The horizontal bar will be too near the top to be attractive. To fix this, simply change the first number in line 20. Where it now says "IF X=6", try "IF X=7" or "IF X=8". As a general rule, it is nice to have an equal number of characters above and on both sides of the point of intersection.

You are not through yet! Try making a double-thick cross. Make two rows and columns of characters. Now you will need an *even* number of characters in the horizontal line. Let your imagination begin to work. What other improvements occur to you? A Celtic ring? A three-tiered base? What about an electronic beep every time a new line is printed? Have you thought of slowing down the printing process with time-delay loops nested inside the main loop? Experiment! Your computer is limited only by your imagination.

Here is an example of this basic program in its elemental form adapted for the TRS-80, Model II. Note that the program uses double figures to give it a little more presence on the screen.

```
100 REM TRS MOD II CROSS
105 CLS
110 FOR X=1 TO 19
120 IF X=6 THEN PRINTTAB(23);"**********************************":GOTO 140
130 PRINTTAB(37);"**"
140 NEXT X
150 END
```

4

PASSWORD: SHIBBOLETH

If you are a beginning programmer, work with this game and the next two before proceeding to the more complicated programs. They will give you a taste of the joys that are to come, and any errors you make will be easy to spot.

If you have not read chapter 2 on adapting the programs, it would be advisable for you to do so now. Important information given in that chapter will not be repeated in the introductions to the games themselves.

ing the Jordan. It seems the Ephraimites could not pronounce it correctly. Once you have played this game a few times, you will never forget the word. It has actually come into our English language as a word meaning a criterion or a test, a word that distinguishes members of a particular group. This computer game challenges the player to "get it right."

Line 185 allows two possible winning combinations. The computer will accept "SHIBBOLETH" either preceded or followed by three spaces. If you wish, you can add other combinations of spaces that will register as wins. Perhaps one before and two after would be a worthy addition. In actual practice this has proven to be unnecessary baggage. To keep the program short and simple, we limited it to the two most probable final orders.

Line 180 puts a number above each letter and space in the jumbled word. This is a tremendous help to the player who must enter each move by typing a number "from" and "to." The final 0123, of course, represent 10, 11, 12, and 13. Some players need to be reminded of this. If you will look at the programs for "Captive!" and "Seven Churches," you will see a method of adding a *1* above the higher numbers. You can insert a line 179 for this purpose if you want. Just enter nine spaces and four ones inside quotation marks.

The INPUT statement in line 190 makes it clear to the player that a comma is required between the "from" and "to" locations.

Line 195 was an afterthought. Some players accidentally entered a number greater or smaller than the game was expecting. This threw everything into disarray. With line 195 in place, the confused computer merely asks, "What?" and waits for a more sensible command. Lines 200 and 210 also prevent impossible moves.

Your attention needs to be directed to the DATA statement in line 1000. Other games will make heavy use of DATA, and the format will vary. Be careful! In this game, a comma separates each letter and space. Copy it *exactly* as printed. Later, if you want, you can rearrange the order of the letters and spaces, but the commas must stay. Put only one comma, please, between each letter and space. It is an easy mistake to hit a period, so watch what you're doing.

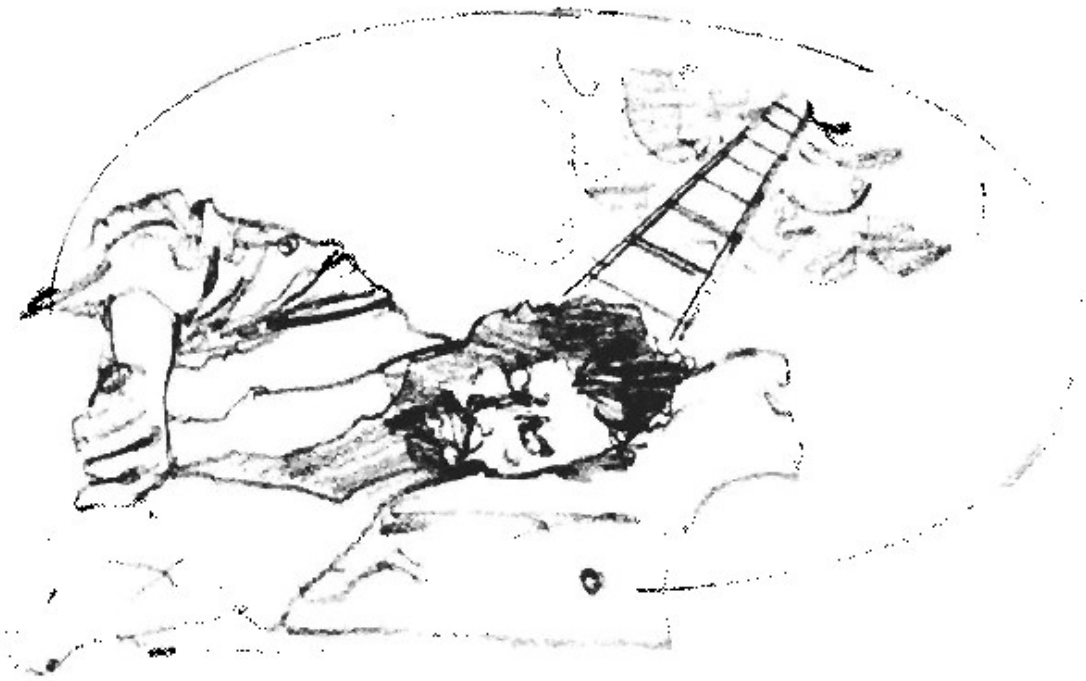
Password: Shibboleth

```
10 REM  "PASSWORD: 'SHIBBOLETH'"
100 PRINT "Ø":REM  REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
110 DIM W$(13)
120 FOR I=1 TO 13
130 READ W$(I)
140 NEXT I
150 FOR I=1 TO 13
160 W$=W$+W$(I)
170 NEXT I
180 PRINT:PRINT "1234567890123":PRINT W$:PRINT
185 IF W$="---SHIBBOLETH" OR W$="SHIBBOLETH---" THEN 300
190 INPUT "FROM,TO";F,T
195 IF F>13 OR T>13 OR F<1 OR T<1 THEN PRINT "WHAT?":GOTO 190
200 IF W$(F)="-" THEN PRINT "CAN'T MOVE FROM SPACE":GOTO 190
210 IF W$(T)<>"-" THEN PRINT "CAN'T MOVE TO LETTER":GOTO 190
220 W$(T)=W$(F):W$(F)="-"
240 W$="":X=X+1:GOTO 150
300 IF X<20 THEN PRINT "GOOD-":PRINT
310 PRINT "YOU GOT IT IN";X
320 PRINT "GUESSES."
330 PRINT:INPUT "TRY AGAIN";W$
340 IF W$="N" OR W$="NO" THEN END
350 RUN
1000 DATA B,I,T,-,S,O,L,-,H,E,B,-,H
```

Dressing up this game for the TRS-80 is simple and easy. Here is a direct printout. Because the printer employed uses a daisy wheel, there are no diagonal slash marks through the zeros. Note that the letter *O* is fat, and the number *0* is skinny. The line numbers do not correspond to those in the previous version.

```
100 REM PASSWORD: 'SHIBBOLETH'
110 REM TRS-80 MOD.][ VERSION BY DAVID BANGLEY
120 CLS
130 DIM W$(13)
140 FOR I=1 TO 13
150 READ W$(I)
160 NEXT I
165 PRINTTAB(20);"**********  SHIBBOLETH  **********"
167 PRINT@(10,0),""
170 FOR I=1 TO 13
180 W$=W$+W$(I)
```

```
220 PRINTTAB(10);W$
230 IF W$="---SHIBBOLETH" OR W$="SHIBBOLETH---" THEN 500
240 PRINT:PRINTCHR$(26)
250 INPUT"FROM,TO";F,T
260 PRINTCHR$(25)
270 IF F>13 OR T>13 OR F<1 OR T<1 THEN PRINTTAB(10);"WHAT?":GOTO 240
280 IF W$(F)="-" THEN PRINTTAB(10);"I CAN'T MOVE FROM A SPACE.":GOTO 240
290 IF W$(T)<>"-" THEN PRINTTAB(10);"I CAN'T MOVE TO A LETTER.":GOTO 240
300 W$(T)=W$(F):W$(F)="-"
310 W$="":X=X+1:GOTO 170
500 IF X<20 THEN PRINTTAB(10);"THAT IS VERY GOOD!":PRINT
510 PRINTTAB(10);"YOU SUCCESSFULLY REARRANGED IT IN";X;"MOVES."
520 PRINT:PRINT
530 PRINTTAB(10);"WOULD YOU LIKE TO TRY AGAIN ?"
540 INPUT W$
550 IF W$="N" OR W$="NO" THEN PRINTTAB(10);"BYE.":END
560 RUN
900 DATA B,I,T,-,S,O,L,-,H,E,B,-,H
999 END
```



5

JACOB'S LADDER

Genesis 28:10–22 relates the familiar story of how Jacob slept with a stone for a pillow. He dreamed of a ladder reaching from earth to heaven, with angels moving up and down it. He was promised some grand things for the future. When he awoke he was deeply impressed with the close presence of the Lord.

This simple computer game uses the ladder as a method of scoring responses to biblical questions. This game is placed early in the book to give you a valuable creative opportunity. There are dozens of easy things you can do to make this program your own.

of answers to oral questions is entered into the computer. The ladder keeps track of the progress of the individual or the class. An asterisk ascends "higher, higher," one rung at a time, with each correct answer. When it reaches "heaven," a waterfall of *Success!* flows down the ladder. Wrong answers kick the asterisk one rung lower. If it hits the bottom there is a warning. Another wrong answer, at the bottom, will lose the game.

Start fixing this one by entering it exactly as printed. You can do that in a jiffy. Do it now and then come back to the text.

Jacob's Ladder

```
10 REM  "JACOB'S LADDER"
100 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
105 A=19
110 PRINT:PRINT:PRINT "JACOB'S LADDER"
120 FOR T=1 TO 1000:NEXT T
130 PRINT "□":REM  REPLACE AS ABOVE.
140 FOR L=1 TO 21
150 PRINT "X";
160 NEXT L
170 PRINT
180 FOR L=1 TO 19
190 IF A=1 THEN PRINT TAB(7);"SUCCESS!":GOTO 220
200 IF A=L THEN PRINT TAB(10);"*":GOTO 220
210 PRINT TAB(10);"="
220 NEXT L
230 FOR L=1 TO 21
240 PRINT "-";
250 NEXT L
260 PRINT
265 IF A=1 THEN END
270 INPUT A$
280 IF A$="T" THEN A=A-1
290 IF A$="F" THEN A=A+1
300 IF A=20 THEN 400
310 GOTO 130
400 B=B+1
410 IF B=2 THEN PRINT "YOUR HIP IS OUT OF JOINT.":END
420 A=19
430 PRINT "WATCH OUT!"
440 FOR T=1 TO 1000:NEXT T
450 GOTO 130
```

RUN the program and debug it. Did you get the colons and semicolons right? What you should see is a row of the letter *X* across the top, a straight ladder of equals signs with an asterisk at the bottom, and a line of dashes for the ground. Your cursor should be blinking, waiting for INPUT.

Now enter a *T*. The asterisk should climb to the next rung. Go ahead and run it to the top and see if you can set off the reward. Now run it again, entering an *F*. You should get a caution message. Enter another *F*. This should get you the losing message and end the game.

If all of that works, you have an error-free program and are ready to go to work. Take a look at the graphics. Does the field of play fill your screen, or is it scrunched in the upper right-hand corner? You can widen "heaven" and make it stretch from one side to the other. Play with the large number in line 140. As it is, it will print twenty-one letters in a row. Change it to thirty-nine and it will stretch across a forty-column screen. Keep trying numbers until you like what you see.

What about the symbol? *X* is rather dull for heaven. Try to make some clouds. This could be a rare opportunity to use your @. Experiment!

Once you have arranged heaven to suit you, go down to earth. Change the larger number in line 230 to equal the one you place in line 140. Now you will have a symmetrical heaven and earth with a ladder too far to the right. Change the TAB numbers in lines 200 and 210 to move it.

The ladder can be made longer or shorter by changing the number in line 230. Too long, and it may become a bore. Too short, and you lose the thrill of accomplishment.

If you are an electronic artist, you will accept another challenge. Try to draw Jacob sleeping at the bottom of the ladder, his head on a stone. Add an angel or two in the open spaces. You can do some wonderful things with the characters on a standard keyboard. If you have special graphics there is virtually no limit to what is possible.

Line 410 ends the game with a failure statement. In this version, I have used, "Your hip is out of joint." That actually happened to Jacob at Peniel when he wrestled an angel. You can read about it in Genesis 32:22–32. I tacked it onto the game because it helps to remind us of another thing about Jacob. In the second version of the game it says, "The dream is over." If you want, you can let it say something like, "The rock ought to be under your head—not in it!"

This version of the program is not actually a game. It is merely a gimmick for graphically displaying progress. The *T* and *F* in lines 280 and 290 represent nothing but right and wrong. A teacher could ask a series of questions and record the results of each on the ladder.

But don't stop here. The program can easily be turned into a full game. You can put all the questions you want in DATA and make the program respond to the correctness of the input. Examine the program below and you'll see how I've started this process for you.

Jacob's Ladder (with sample questions)

```
10 REM  "JACOB'S LADDER" WITH SAMPLE QUESTIONS.
100 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
105 A=19
110 PRINT:PRINT:PRINT "JACOB'S LADDER"
120 FOR T=1 TO 1000:NEXT T
130 PRINT "□":REM  REPLACE AS ABOVE.
140 FOR L=1 TO 21
150 PRINT "X";
160 NEXT L
170 PRINT
180 FOR L=1 TO 16
190 IF A=1 THEN PRINT TAB(7);"SUCCESS!":GOTO 220
200 IF A=L THEN PRINT TAB(10);"*":GOTO 220
210 PRINT TAB(10);"="
220 NEXT L
230 FOR L=1 TO 21
240 PRINT "-";
250 NEXT L
260 PRINT
265 IF A=1 THEN END
267 READ Q$,A$
268 PRINT Q$
270 INPUT D$
280 IF D$=A$ THEN A=A-1:GOTO 300
290 A=A+1
300 IF A=17 THEN 400
310 GOTO 130
400 B=B+1
410 IF B=2 THEN PRINT "THE DREAM IS OVER.":END
420 A=16
430 PRINT "WATCH OUT!"
440 FOR T=1 TO 1000:NEXT T
450 GOTO 130
1000 DATA "JESUS GREW UP IN NAZARETH.",T
1010 DATA "PAUL WAS A SADDUCEE.",F
1020 REM ADD MORE DATA AS EXPLAINED IN TEXT.
```

Look first at the last three lines. In line 1000 you are given an example of a Bible question with *True* for an answer. Line 1010 demonstrates a *False* answer. You can add as many true/false questions as you like. Place each one on a new DATA line, with quotation marks around the statement. Place the correct answer, *T* or *F*, after

questions. The computer will ask each one in turn until the player wins, loses, or runs out of DATA.

If you want to add some trick questions, try these:

"The Bible says there were three wise men." (It doesn't.)

"The Bible says Calvary is a hill." (It doesn't. It says it is a "place." Only a recent paraphrase of the Bible and some hymns say it is a hill.)

"The Bible says Jesus fell under the weight of the Cross." (It doesn't.)

"The Bible says, 'Heaven helps those who help themselves.' " (It doesn't.)

Common assumptions like these are good discussion—or argument!—starters. The computer will not take sides.

Notice how the ladder has been shortened to sixteen rungs in line 180. The same limit is given the variable *A* in lines 105 and 420. This allows the question to appear at the bottom of the screen within the vertical limits of the first program. Again, you can make your ladder of any length.

Observe the way additional lines have been tucked into the spaces left blank in the first program. Line 267 adds READ Q$, A$. This instructs the computer to read the DATA and create a *Q*uestion string and an *A*nswer string. Line 268 prints the question string. Line 270 has been changed to rename the INPUT D$. Now the computer has something to look for. If the player's D$ is the same as the computer's A$, then you have a winner. The A = A − 1 actually kicks the asterisk *up* a notch. The GOTO 300 skips over the wrong answer penalty in line 290.

You may want to add a notation in line 270 that tells the player what the computer is expecting as INPUT. It could say, INPUT 'T OR F';D$. On the screen you will see: "T OR F?" and a flashing cursor.

Continue making alterations until you can't think of anything else to do. This kind of computer doodling is a great way to learn your way around BASIC. For an example of some of my own doodling, see the introduction I prepared for my son David's "Bible Pairs."

6

SHEEP AND GOATS

Matthew 25:31–46 is an often-quoted passage that begins with Jesus telling how he will "gather all the nations, and . . . separate them one from another as a shepherd separates the sheep from the goats." He says that he will place the sheep at his right hand and the goats at his left.

This easy program makes a difficult challenge of that sorting. You are given six sheep and six goats, represented by *O* and *X*. A random integer generator places them in a line containing one blank space. The player must move one animal at a time to the blank space

Line 230 prevents the random generator from ever accidentally doing this for you.

The same "from, to" technique used in "Password: Shibboleth" is used in this game, but now it is more restricted. There is only one blank space, and the player can move forward or backward only one or two spaces. That is, it is possible to move an animal one space right or left directly into the blank, or it is possible to jump over one animal right or left in order to land in it. Jumps larger than this are prevented by line 290.

Lines 110 through 180 place the symbols in a fresh arrangement each game. Lines 260 through 290 tell the player an illegal move has been requested. Line 300 is the formula for a correct move.

Don't let this deceptively simple program mislead you. It can be an absorbingly difficult game. More than fifty moves are sometimes necessary to place everything in the correct position.

Sheep and Goats

```
10 REM  "SHEEP AND GOATS"
50 PRINT "J":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
60 PRINT:PRINT "X = GOAT"
70 PRINT:PRINT "O = SHEEP"
80 PRINT:PRINT "WIN = XXXXXX-OOOOOO"
90 PRINT:PRINT
100 DIM C$(13)
110 FOR X=1 TO 3
120 FOR I=1 TO 6
130 R=INT(RND(1)*13)+1
140 IF C$(R)<>"" THEN 130
150 IF X=1 THEN C$(R)="X"
160 IF X=2 THEN C$(R)="O"
170 IF X=3 THEN C$(R)="-":GOTO 200
180 NEXT I:NEXT X:X=0
200 FOR I=1 TO 13
210 C$=C$+C$(I)
220 NEXT I
230 IF X=0 AND C$="XXXXXX-OOOOOO"THEN RUN
240 PRINT:PRINT "1234567890123":PRINT C$:PRINT
245 IF C$="XXXXXX-OOOOOO" THEN 500
250 PRINT:INPUT "FROM,TO";F,T
260 IF F>13 OR F<1 OR T>13 OR T<1 THEN PRINT ">WHAT?":GOTO 250
270 IF C$(F)="-" THEN PRINT ">CAN'T MOVE FROM SPACE":GOTO 250
280 IF C$(T)<>"-" THEN PRINT ">CAN'T MOVE TO ANIMAL":GOTO 250
290 IF ABS(T-F)>2 THEN PRINT ">JUMP TOO LARGE":GOTO 250
300 C$(T)=C$(F):C$(F)="-"
310 X=X+1:C$="":GOTO 200
500 PRINT "YOU WON IN";X;"MOVES"
510 IF X<20 THEN PRINT "EXCELLENT!"
520 IF X>19 AND X<40 THEN PRINT "GOOD."
530 PRINT:INPUT "TRY AGAIN";C$
540 IF C$="N" OR C$="NO" THEN END
550 RUN
```

7

PEOPLE WHO MET CHRIST

In this game the computer selects the name of a person mentioned in the Gospels who came face-to-face with Jesus Christ. The first letter of the name is displayed on the screen and the computer waits for either a guess or a request for the next letter.

When you ask for the next letter, the distinctive core of this program comes into play. You will find it in lines 340 and 350. The variable *L* in line 340 is the length of the name selected. It counts the number of letters in the person's name. In line 350 the top score of one hundred is divided by the number of letters. The resulting

requested. A truly perceptive player will be able to work a little mental algebra and determine whether the mystery name is short or long.

It is risky to guess the name too soon. For instance, if you see a *J,* it could be James, Judas, Joseph, or John. A successful gambler, though, will get the highest scores.

The DATA contain twenty names as printed. You may add more names if you wish. If you do, remember these important details: the numbers in lines 100, 110, and 310 must equal the total number of names stored in DATA; and each name must be separated by a comma. If you include a compound name, like The Rich Young Ruler, it must be enclosed within quotation marks. Spaces alone are too weak to hold a name together when a DATA statement is read. The quotation marks cause the computer to read the name as a unit.

Line 330 flags each name as it is used so that there will be no repetition. The flag is caught by line 320, which commands the random generator to make another choice if the name it settles on has already been used.

Line 410 is unique to this game. Notice the space after the bracket that points out the line the computer is printing its choice on. LEFT$ causes each letter to be printed in sequence from left to right.

Lines 425 and 895 nullify the A$ variable. This prevents a name like Nicodemus being read as "No," thus ending the game. Incidentally, you can see where this program was fine-tuned after its creation by the numbering of the lines. Note especially what happens between lines 300 and 310 so that a reading of the current score could be added. You can continue the process by tucking in any idea that occurs to you. The spaces were left for that purpose.

Some brands of computer may require a space inside the quotation marks in line 630 so that the number of points being added to your score will not run into the language on either side. Other computers add the spaces automatically.

By all means you should clean up the title and instructions for this game. The points listed are things that need to be remembered if you load the program from a cassette many months after you first create it. Adjust the spacing. Add a little sparkle.

One person who played this game complained about the spelling of a couple of the names. She was sure that Nathanael ought to be Nathaniel, and that Barabbas was Barabbus. Boo on that last one! Certainly Mr. Hawthorne spelled his name Nathaniel. But each of these names was copied directly from the King James Bible. If the

DATA the way you prefer. Once you have entered it, there is no court of appeal. The computer will accept only perfect spelling as determined by its own internal dictionary.

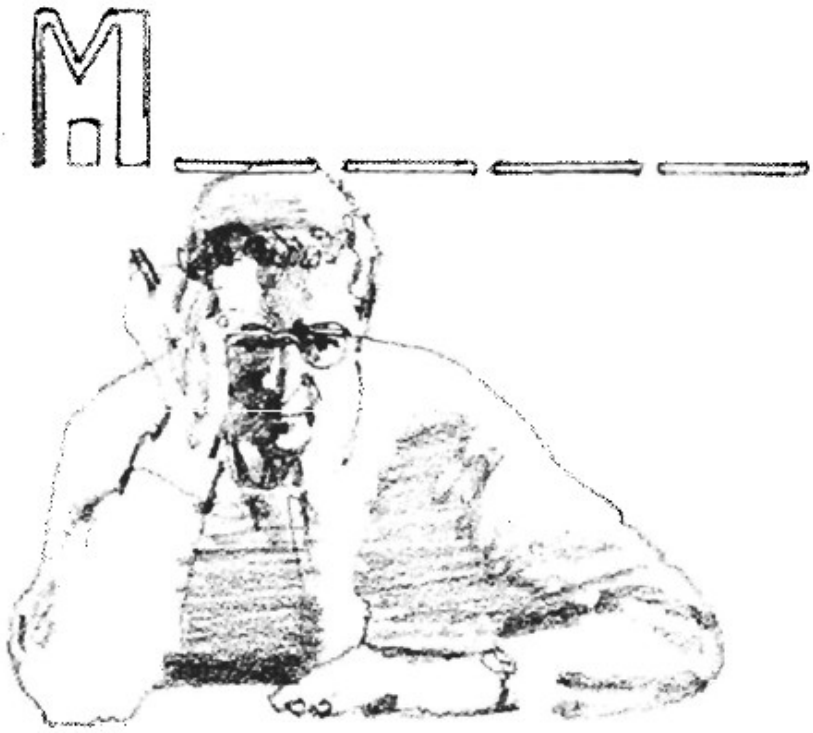
The game is set up to provide five names per round. Change the final number in line 300 for whatever else you might want. A good way to expand this game would be to give some kind of super reward if the player completes four perfect rounds.

Have fun! Don't tell anyone that what they are really doing is learning to spell *Caiaphas*.

People Who Met Christ

```
10 REM "PEOPLE WHO MET CHRIST"
100 DIM N$(20),F(20)
110 FOR L=1 TO 20
120 READ N$(L)
130 NEXT L
140 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
150 PRINT "PEOPLE WHO MET CHRIST"
160 PRINT
170 PRINT "I WILL SPELL THE NAMES OF SOME PEOPLE IN THE LIFE OF CHRIST. . ."
180 PRINT "ONE LETTER AT A TIME.":PRINT
190 PRINT "YOU TRY TO GUESS WHO I AM THINKING OF BEFORE I FINISH IT."
200 PRINT
210 PRINT "WHEN YOU THINK YOU KNOW WHO IT IS, TYPE IN YOUR GUESS."
220 PRINT "IF YOU NEED MORE HELP, PRESS RETURN FOR ANOTHER LETTER."
230 PRINT
240 INPUT "PRESS RETURN TO START";A$
250 PRINT "□":REM  REPLACE AS ABOVE.
300 FOR D=1 TO 5
304 S=S+V
305 PRINT "CURRENT SCORE:";S
307 PRINT
310 W=INT(RND(1)*20)+1
320 IF F(W)=1 THEN 310
330 F(W)=1
340 L=LEN(N$(W))
350 Q=INT(100/L)
360 V=100
370 FOR R=1 TO L-1
380 PRINT "LETTER #";R
390 PRINT "POINT VALUE:";V
400 PRINT
410 PRINT "] ";LEFT$(N$(W),R)
420 PRINT
425 A$=""
430 INPUT">";A$
440 IF A$<>"" THEN 500
450 PRINT
460 V=V-Q
470 NEXT R
475 V=0
480 PRINT "SORRY, YOU LOOSE!"
490 GOTO 600
500 IF A$=N$(W)THEN 600
510 V=0
520 PRINT "WHOOPS, WRONG PERSON!"
530 GOTO 600
600 PRINT "□":REM  REPLACE AS ABOVE.
610 PRINT "CORRECT!"
620 PRINT
```

```
630 PRINT "THAT ADDS";V;"POINTS TO YOUR SCORE."
640 PRINT
660 GOTO 860
800 PRINT "THE NAME WAS:"
810 PRINT N$(W)
820 PRINT
830 INPUT "ANOTHER WORD";A$
840 IF LEFT$(A$,1)="N" THEN PRINT "BYE.":END
850 PRINT "J":REM  REPLACE AS ABOVE.
860 NEXT D
870 PRINT"THIS ROUND IS OVER."
880 PRINT
885 PRINT "FINAL SCORE:";S
890 PRINT:PRINT "WOULD YOU LIKE ANOTHER ROUND?"
895 A$=""
900 INPUT A$
910 IF LEFT$(A$,1)="N" THEN PRINT "OKAY, BYE.":END
920 RUN
930 DATA MARY,JOSEPH,HEROD,NICODEMUS,SIMON
940 DATA ZACCHAEUS,MARTHA,LAZARUS,ANDREW,JAMES
950 DATA JOHN,PHILIP,NATHANAEL,THOMAS,MATTHEW
960 DATA JUDAS,ANNAS,CAIAPHAS,PILATE,BARABBAS
```



8

THINGS IN THE BIBLE

Here is an original rendition of a game familiar to many as "Hangman." The concept was a popular parlor game generations ago. A secret word was chosen and the players tried to guess it one letter at a time. Each wrong guess let the leader draw a portion of a scaffold. If the players discovered the word before the scaffold could be completed, they won.

There are a half-dozen approaches to playing "Hangman" on a computer. The beauty of this one is its compactness. "Things in the Bible" presents a bare-bones approach to the game. The program

that follows in the next chapter, "Where Are We Now?" is a deluxe edition that adds many very nice touches.

The DATA bank contains forty-nine (count them!) names of things that are mentioned in the Bible. Some are easy; some are quite difficult. Most players have a hard time with *myrrh*. If you change the contents of the DATA statements, be sure you correct the number in parentheses in line 190. Adhere strictly to the format of the DATA by placing a comma after everything except the end of a line and by placing an asterisk between words. The program uses these asterisks as signposts along the way.

In this version there is no penalty for wrong guesses. The player can try every letter on the keyboard until the mystery word is figured out. In practice, however, a player who isn't told this will make very careful guesses. I watched one woman devote twelve minutes to figuring out the word *ivory*.

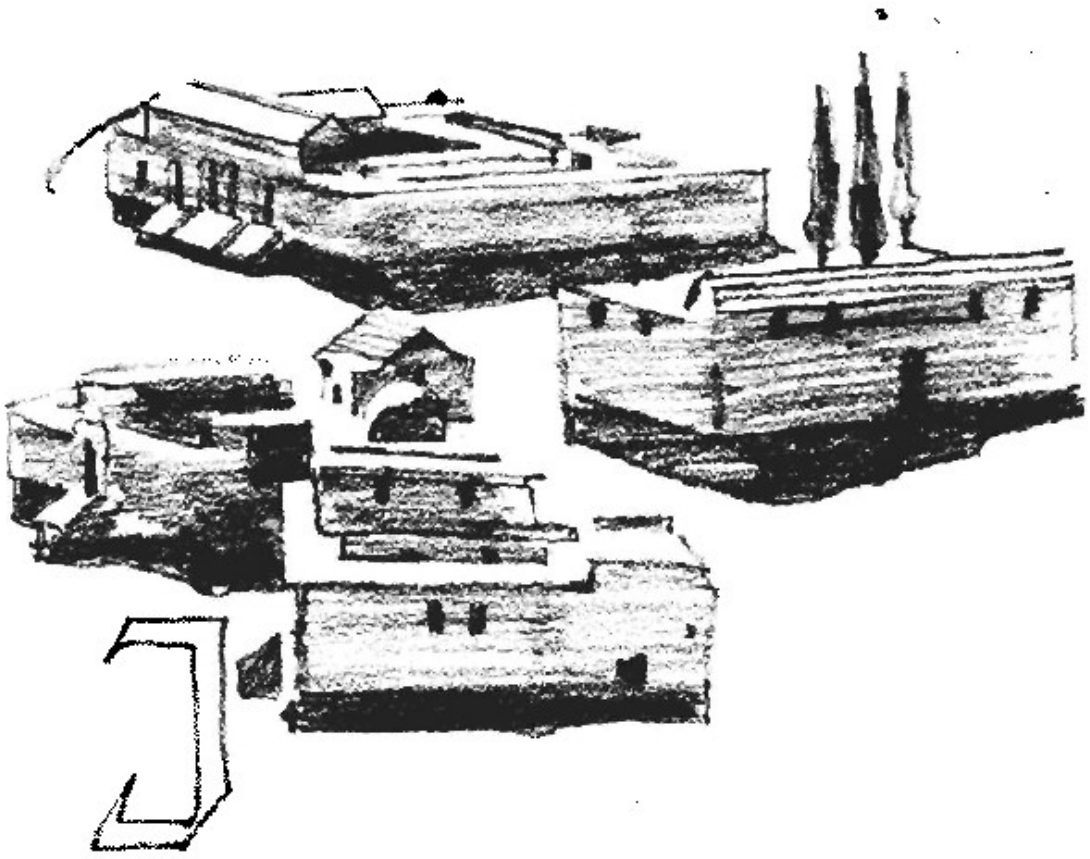
The game is one giant loop beginning at line 200 and flipping back in line 540. The way it is printed here, there is a limit of ten mystery words to each game, all of which will be correctly guessed if the player will stick with it. Change the number in line 200 and you change the number of words per game.

Notice the space after the last colon in line 350. If you omit it, the name of the secret thing will run into the PRINT statement. Spacing is very important, but hard to convey in a published printout. Look at your results on the screen. Go back through your program and do some housekeeping.

Things in the Bible

```
10 REM  "THINGS IN THE BIBLE"
100 PRINT "J":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
110 PRINT:PRINT:PRINT "I WILL CHOOSE A THING"
120 PRINT "MENTIONED IN THE BIBLE."
130 PRINT:PRINT "YOU GUESS IT ONE"
140 PRINT "LETTER AT A TIME."
150 PRINT:PRINT
190 DIM F(49)
200 FOR G=1 TO 10
210 RESTORE:S=0
220 D=D+1
230 RN=INT(RND(1)*49)+1
240 IF F(RN)=1 THEN 230
250 F(RN)=1
260 READ L$
270 IF L$="*" THEN S=S+1
280 IF S=RN THEN 300
290 GOTO 260
300 S=0
310 READ L$:S=S+1
320 IF L$="*" THEN 340
330 L$(S)=L$:GOTO 310
340 S=S-1
350 PRINT:PRINT:PRINT "THE THING: ";
360 FOR X=1 TO S
370 IF G$(X)=L$(X) THEN PRINT G$(X);:GOTO 390
380 PRINT "-";
390 NEXT X
400 FOR X=1 TO S
410 IF G$(X)="" THEN 430
420 NEXT X
425 GOTO 500
430 PRINT:PRINT
440 INPUT "YOUR GUESS";G$
450 PRINT:PRINT
460 FOR X=1 TO S
470 IF G$=L$(X) THEN G$(X)=L$(X)
480 NEXT X
490 GOTO 350
500 PRINT:PRINT:PRINT "YES! YOU GOT IT!"
510 FOR X=1 TO 10
520 G$ (X)="":L$(X)=""
530 NEXT X
540 NEXT G
550 PRINT:PRINT "THAT'S 10 THINGS..."
560 PRINT:PRINT "FOR 10 MORE, TYPE: RUN"
1000 DATA *,A,L,T,A,R,*,A,N,G,E,L,*,A,R,K,*,A,R,M,O,R,*,B,A,R,L,E,Y,*
1010 DATA B,E,A,R,D,*,B,E,H,E,M,O,T,H,*,B,R,I,M,S,T,O,N,E,*,C,A,M,E,L,*
1020 DATA C,H,A,R,I,O,T,*,E,P,H,O,D,*,E,P,I,S,T,L,E,S,*,F,I,G,*,H,Y,S,S,O,P,*
1030 DATA I,D,O,L,*,I,V,O,R,Y,*,L,A,M,P,*,L,E,A,V,E,N,*,L,E,G,I,O,N,*
```

```
1050 DATA O,F,F,E,R,I,N,G,S,*,O,I,N,T,M,E,N,T,*,O,L,I,V,E,*
1060 DATA P,H,Y,L,A,C,T,E,R,Y,*,P,S,A,L,T,E,R,Y,*,S,A,P,P,H,I,R,E,*
1070 DATA S,C,O,U,R,G,E,*,S,E,P,U,L,C,H,E,R,*,S,H,E,E,P,F,O,L,D,*
1080 DATA S,P,A,R,R,O,W,*,M,Y,R,R,H,*,S,T,O,N,E,*,S,Y,N,A,G,O,G,U,E,*
1090 DATA S,Y,C,A,M,O,R,E,*,T,A,B,E,R,N,A,C,L,E,*,T,A,L,E,N,T,*,T,A,X,E,S,*
1100 DATA T,E,M,P,L,E,*,T,H,R,O,N,E,*,T,H,U,M,M,I,M,*,T,I,M,B,R,E,L,*
1110 DATA T,I,T,H,E,*,T,R,U,M,P,E,T,*,U,R,I,M,*
1120 DATA V,I,N,E,Y,A,R,D,*,W,I,L,D,E,R,N,E,S,S,*
```



9

WHERE ARE WE NOW?

In this game you have a luxury edition of the same general idea that is the heart of "Things in the Bible." There are vast differences in the degree of refinement.

The problem is to guess the name of a place mentioned in the Bible, one letter at a time. If you make a correct choice, the computer will show you where the letter belongs in the word. If you make an incorrect choice, you lose a turn. It is exciting to see the blank spaces gradually turn into a recognizable name.

Let's look at some of the special features of this game. First of

against another person. If you decide you want to play alone with the names stored in the DATA bank, the program sends the game to line 230, which starts the random selection process. If you prefer to play a companion, the program kicks down to line 350 and asks the other player to enter a word into the computer's memory, one letter at a time. Line 410 limits the number of letters in the mystery word to ten. An asterisk is used to signal the end of the word if it contains fewer than ten letters. Lines 280, 330, and 430 watch for this clue. If you examine the DATA statements you will notice that an asterisk is used there to signal the division between words.

If the player accidentally enters a number other than 1 or 2 for the initial choice, line 200 will ask, "What?" This is a venerable computer routine that never fails to please.

If a human source of the mystery word makes a mistake, it is necessary only to enter the word *error* in order to restart the sequence.

The variable LT holds unsuccessful attempts at guessing the correct letter. You can see it at work in lines 520 and 530. The code not only keeps count of the number of wrong guesses, it also prints out a list of them at the top of the screen. Notice how it works in lines 660 and 670.

Lines 690 through 710 do some especially pleasing things on the screen for the player. Keeping the player fully informed is always a good idea. Notice how line 700 actually counts the number of times a certain letter appears in the mystery word. The variable is *Z*.

As the game is set up, the player is allowed six wrong guesses. This seems to be about right, but you can change it any way you like by altering line 810 and changing the language in lines 850 and 1090.

Entering the DATA is easy, but you must be careful. Notice that you must start with an asterisk followed by a comma. Each letter of the mystery word must be separated by a comma. The only place you should not put a comma is after the last asterisk in each line of DATA. The most common mistake in entering DATA of this sort is the accidental placement of two commas where you thought you put only one. Double-check each line as you enter it. Remember, you must use commas; periods and semicolons will not do.

The list in the printout contains thirty-nine cities and places mentioned in the Bible. You can add to the list, or replace it entirely. Count the number of places in your DATA bank and enter that number in the parentheses in line 100. Also add the same number to the random integer selector in line 240. Line 940 will also need

This game will work with a group, but it is at its best with one or two players. There is an intimate quality to the relationship between the players and the computer that becomes very pleasant.

Where Are We Now?

```
10 REM  "WHERE ARE WE NOW?"
100 DIM F(39)
110 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
120 PRINT "   WHERE ARE WE NOW?"
130 PRINT:PRINT:PRINT "OPTIONS:"
140 PRINT:PRINT "1) PLAY MY DATA BANK"
150 PRINT:PRINT "         -OR-"
160 PRINT:PRINT "2) PLAY SOMEONE ELSE"
170 PRINT:PRINT:PRINT:INPUT "YOUR CHOICE";S
180 IF S=1 THEN 230
190 IF S=2 THEN 350
200 PRINT:PRINT "WHAT?"
210 GOSUB 1030
220 GOTO 110
230 S=0:RESTORE:D=D+1
240 RI=INT(RND(1)*39)+1
250 IF F(RI)=1 THEN 240
260 F(RI)=1
270 READ L$
280 IF L$="*" THEN S=S+1
290 IF S=RI THEN 310
300 GOTO 270
310 S=0
320 READ L$:S=S+1
330 IF L$="*" THEN 470
340 L$(S)=L$:GOTO 320
350 PRINT "□":REM  REPLACE AS ABOVE
360 PRINT "ENTER YOUR WORD ONE"
370 PRINT "LETTER AT A TIME."
380 PRINT:PRINT "USE * TO END WORD."
390 PRINT:PRINT "IF YOU MAKE A MISTAKE"
400 PRINT "TYPE 'ERROR'.":PRINT:PRINT
410 FOR S=1 TO 10
420 PRINT:INPUT "LETTER";L$
430 IF L$="*" THEN 470
440 IF L$="ERROR" THEN RUN
450 L$(S)=L$
460 NEXT S
470 S=S-1:GOSUB 1050
480 PRINT:PRINT "HIT RETURN TO START."
490 INPUT Q$
500 PRINT "□":REM  REPLACE AS ABOVE
510 PRINT "LETTERS USED-";
520 FOR X=1 TO LT
530 PRINT LT$(X);
540 NEXT X
550 PRINT:PRINT:PRINT "WRONG GUESSES:";LT
560 PRINT:PRINT "WORD> ";
570 FOR X=1 TO S
580 IF Q$(X)=L$(X) THEN PRINT Q$(X);:GOTO 600
```

```
600 NEXT X
610 PRINT:PRINT:INPUT "GUESS";G$:PRINT:PRINT
620 Z=0
630 FOR X=1 TO S
640 IF G$=L$(X) THEN G$(X)=L$(X):Z=Z+1
650 NEXT X
660 FOR X=1 TO LT
670 IF G$=LT$(X) THEN PRINT "YOU HAVE USED '";G$;"'.":GOTO 720
680 NEXT X
690 IF Z=1 THEN PRINT "THERE IS ONE ";G$
700 IF Z>1 THEN PRINT "THERE ARE";Z;G$;"'S"
710 IF Z=0 THEN PRINT "SORRY,NO ";G$;" IN WORD.":LT=LT+1:LT$(LT)=G$
720 GOSUB 1030
730 FOR X=1 TO S
740 IF G$(X)="" THEN 810
750 NEXT X
760 PRINT "□":REM  REPLACE AS ABOVE
770 PRINT "YOU ARE CORRECT, THE"
780 PRINT "WORD WAS ";
790 GOSUB 960
800 PRINT ".":PRINT:GOTO 900
810 IF LT=6 THEN 830
820 GOTO 500
830 PRINT "□":REM REPLACE AS ABOVE
840 PRINT "SORRY, BUT YOU HAVE"
850 PRINT "USED YOUR 6 WRONG"
860 PRINT "GUESSES."
870 PRINT:PRINT "THE WORD WAS ";
880 GOSUB 960
890 PRINT ".":PRINT
900 PRINT "DO YOU WANT TO PLAY"
910 INPUT "AGAIN";G$
920 IF G$="NO" OR G$="N" THEN PRINT "BYE.":END
930 GOSUB 990
940 IF D=39 THEN PRINT "DATA EXHAUSTED":END
950 GOTO 110
960 FOR X=1 TO S
970 PRINT L$(X);
980 NEXT X: RETURN
990 FOR X=1 TO 10
1000 G$(X)="":L$(X)="":LT$(X)=""
1010 NEXT X
1020 LT=0:RETURN
1030 FOR X=1 TO 2000
1040 NEXT X: RETURN
1050 PRINT "□":REM  REPLACE AS ABOVE
1060 PRINT "INSTRUCTIONS:"
1070 PRINT:PRINT "ENTER GUESSES ONE"
1080 PRINT "LETTER AT A TIME."
1090 PRINT:PRINT "6 WRONG GUESSES END"
1100 PRINT "GAME.":RETURN
2000 DATA *,A,N,T,I,D,C,H,*,A,R,A,B,I,A,*,A,R,A,R,A,T,*,B,A,B,Y,L,O,N,*
2010 DATA B,E,T,H,A,N,Y,*,B,E,T,H,E,L,*,B,E,T,H,L,E,H,E,M,*
2020 DATA B,E,T,H,S,A,I,D,A,*,C,A,N,A,A,N,*,C,A,P,P,A,D,O,C,I,A,*
2030 DATA C,O,R,I,N,T,H,*,C,Y,P,R,U,S,*,D,A,M,A,S,C,U,S,*,E,D,E,N,*
2040 DATA E,G,Y,P,T,*,E,M,M,A,U,S,*,G,A,Z,A,*,H,E,B,R,O,N,*
2050 DATA J,E,R,U,S,A,L,E,M,*,J,O,R,D,A,N,*,J,U,D,E,A,*,K,E,D,E,S,H,*
2060 DATA L,E,B,A,N,O,N,*,M,A,C,E,D,O,N,I,A,*,M,E,G,I,D,D,O,*
2070 DATA M,I,Z,P,A,H,*,N,A,Z,A,R,E,T,H,*,O,L,I,V,E,T,*
2080 DATA P,A,L,E,S,T,I,N,E,*,P,E,R,S,I,A,*,R,A,M,A,H,*,R,O,M,E,*
2090 DATA S,A,M,A,R,I,A,*,S,H,E,C,H,E,M,*,S,H,I,L,O,H,*,S,O,D,O,M,*
2100 DATA T,E,K,O,A,*,T,I,B,E,R,I,A,S,*,T,Y,R,E,*
```

10

BIBLE QUIZ CODE

Originally intended as an introduction to other games, this program can easily be expanded into a complete game in itself. It is a superb teacher. I watched several people play it who knew absolutely nothing about the answers. The best they could do was make hesitant guesses. Invariably they were wrong. But then the computer gave them the correct answer and asked them to start again. Next time, they knew the answer.

The excitement mounts because the game insists that the numbers be entered in the proper sequence, just as you must do when

answered and the fourth is missed, the player is hustled back to the beginning to start over. By the time all seven questions have been answered, the rankest beginner is confident there are sixty-six books in the Bible and all the rest. It is a thrilling process to watch.

Seven questions seem about right if this program is used as an introduction to a larger program. If you want to turn it into an independent game, all you need to do is add some more questions and rework the wording at both ends. You should be warned, however, that if you have a game, say, of twenty-four questions, and your player blows it at number twenty-three, there is likely to be a display of frustration. Is your computer insured?!

To add more questions, just put the three essential elements in additional lines of DATA. First include the numeric answer, then the question, and finally the correct answer/clue. Notice the use of commas and quotation marks in the DATA. Change the number at the start of the main loop in line 250 to equal the total number of questions in DATA. As printed, it reads "7," which means that the loop will cycle back from line 520 seven times before it lets the player into the final part of the program. You can make it cycle as many times as you like, provided you add a new line of DATA for each time around.

Notice that only a *part* of the question is in the second element of DATA. To conserve memory and save your fingers, the regularly repeated "How many" has been assigned in line 100 to H$. Don't miss the space after the "Y." Line 320 then joins H$ to Q$ to make a complete question.

What is Q$? Each DATA statement is divided into three parts. You can see how they are labeled in line 300: A, Q$, R$. The numeric answer is A, the specific question is Q$, and the right answer/clue is R$.

This game is deceptively brief. The original version we came up with used a multitude of IF THEN statements and was almost twice as long as this one. I find entering repeated lines a boring chore. David thinks they are inelegant. By placing the questions and answers in DATA, we eliminated a lot of lines and many possibilities for bugs. Placing the often-used delay loop in the subroutine in lines 600 through 620, we were able to tighten things considerably.

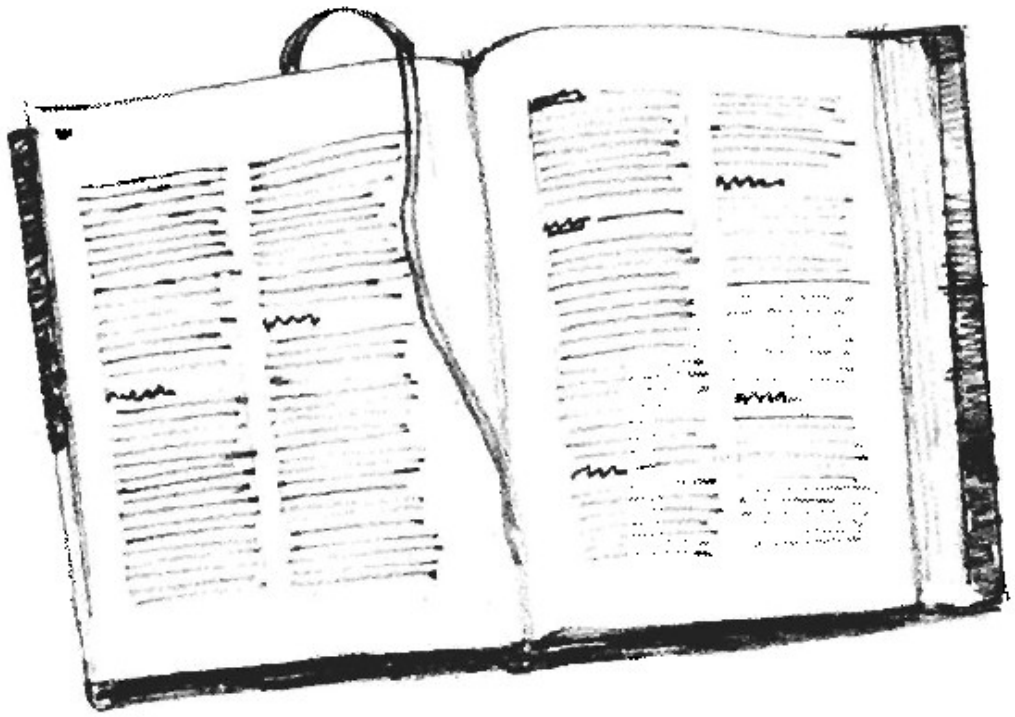
Line 580 is in place as things now stand to keep the computer from going haywire at the completion of this program's routine. If you are planning to go on to another game ("Reverse Which Book?" is recommended), change this line to something like GOTO 1000.

so that 100 becomes 1100, 110 becomes 1110, and so on. This way you can leap right over all the subroutines and DATA and start fresh with the new program.

Don't neglect this little game! It is simple and quick to key into your computer, but the results are extremely engaging. I have heard much laughter and many loud groans from this game's victims. Best of all, it is a strict, but patient teacher. The slowest wit will rapidly learn its DATA. I have yet to see anyone give up before mastering it, and each return to the first question drills in the important facts about the Bible.

Bible Quiz Code

```
10 REM  QUIZ CODE
20 REM  TO BE USED AS GAME INTRODUCTION, OR MODIFED AS GAME.
100 H$="HOW MANY "
110 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
120 PRINT:PRINT:PRINT "BEFORE I PLAY WITH YOU, YOU MUST"
130 PRINT "PROVE THAT YOU ARE A WORTHY OPPONENT."
140 GOSUB 600
150 PRINT:PRINT "THE CODE TO UNLOCK THIS GAME IS"
160 PRINT "STORED IN CORRECT ANSWERS TO"
170 PRINT "SEVEN PRELIMINARY QUESTIONS."
180 GOSUB 600
190 PRINT:PRINT "WRONG ANSWERS--NO GAME"
200 GOSUB 600
210 PRINT:PRINT
220 INPUT "ARE YOU READY";R$
230 PRINT "□":REM  REPLACE AS ABOVE.
240 IF LEFT$(R$,1)="N" THEN 110
250 FOR Q=1 TO 7
300 READ A,Q$,R$
310 PRINT "QUESTION #";Q
320 PRINT:PRINT H$;Q$
330 INPUT R
340 IF R=A THEN 500
350 PRINT:PRINT R$
360 GOSUB 600
370 RESTORE
380 PRINT:PRINT ">CODE MUST BE ENTERED IN SEQUENCE."
390 GOSUB 600
400 PRINT:PRINT "I SEND YOU BACK TO QUESTION #1."
410 PRINT:PRINT "START OVER. . ."
420 GOSUB 600
430 PRINT "□":REM  REPLACE AS ABOVE.
440 GOTO 250
500 PRINT "** CORRECT **":PRINT:PRINT
510 GOSUB 600
520 NEXT Q
530 PRINT "□":REM  REPLACE THIS LINE  AS ABOVE.
540 PRINT "YOU HAVE TRIGGERED MY ELECTRONIC CODE."
545 GOSUB 600
550 PRINT:PRINT "YOU ARE A WORTHY OPPONENT."
560 GOSUB 600
570 PRINT:PRINT "> FILE OPEN"
580 END:REM  SEE TEXT FOR INSTRUCTIONS.
600 FOR T=1 TO 1500
610 NEXT T
620 RETURN
700 DATA 66,"BOOKS IN THE BIBLE","WRONG! THERE ARE 66."
710 DATA 39,"BOOKS IN THE OLD TESTAMENT","ERROR! THERE ARE 39."
720 DATA 27,"BOOKS IN THE NEW TESTAMENT","OUCH! THERE ARE 27."
730 DATA 150,"PSALMS ARE THERE","BAD MUSIC! THERE ARE 150 PSALMS."
740 DATA 1,"CHAPTERS IN PHILEMON","THERE IS ONLY ONE!"
```



11

WHICH BOOK?

This fascinating game is a tremendous aid in learning the order and proper spelling of the books of the Bible. The computer chooses a book at random and keeps it a secret until the player guesses correctly or runs out of turns. This sophisticated program will catch the slightest spelling error. If the player puts an *L* too many in Colossians or tacks on an *S* after Revelation, the computer will quickly protest and ask for another entry. By a series of taunts and cheers the program actually guides the player to the right answer.

Since your computer has memory capacity with nothing in it,

book of the Bible. This becomes the DATA bank upon which the game draws. It is *absolutely essential* that you enter the information accurately. The spelling and titles used in the printed program were copied from the table of contents of the King James Bible. If you want to work with another version, it will be necessary for you to copy that version's spelling. For instance, "Song of Solomon" may be "Song of Songs"; "1 Kings" may be "I Kings." Whatever you enter will be considered correct by your computer. Even the spaces between words are read by the computer with terrible accuracy. So be careful! Make your DATA precisely correct. (I'll bet you mess up Habakkuk!)

A common error in entering DATA is the failure to enclose each title in quotation marks. The next most frequent fumble is to put something other than a comma between the pairs of quotation marks. Anything else—a period, a semicolon, a space—anything else at all, and the program will not RUN. Check and double-check. Let another person proofread for you.

The game's difficulty will be in proportion to the player's familiarity with the Bible. For beginners it is permissible to look at a list of the books while playing. It is fun even with a crutch. But the supreme joy, of course, comes in winning a game with one's own knowledge.

You can give each player as many chances to guess the correct book as you like. Just change the number in lines 140 and 180. Too few chances, and the game is frustrating; too many, and it is a bore. It is better to lose and to start over with a new randomly selected book. Ten to fifteen chances seem to be about right for most people's attention spans, but as you will read in a moment, the lower number is probably better.

Incidentally, if you want to know right away which book the computer has selected, add line 185, PRINT R. This will give you the number of the book. You may find this useful for testing the program, but it should be deleted before you let people start playing the game.

The clues are given in lines 380 through 470. If line 310 senses that the answer is correct, all of these are skipped and the program leaps to line 500. There is nothing sacred about any of the clues. Make up wording of your own. Add extra lines if you like. Simply change the numbers to suit yourself. A clue such as "Try New Testament" or "Try toward Genesis" might be helpful.

It is important for the computer to reveal its secret once a game

spelled, you will be exposed here! Get it correct before you turn it loose on others.

If you can add graphics with your computer, an introduction showing a Bible, perhaps with an animated opening of the cover and a big question mark popping up, could be effective.

An example of character string input (A$) is given for you to use in this game. It is entered in line 110 and can be eliminated entirely or transferred intact to another game. Of course, if you drop it, you must also fix every subsequent line that asks for the A$.

Which Book?

```
10 REM  "WHICH BOOK?"
100 PRINT "HELLO! WHAT'S YOUR"
110 INPUT "NAME";A$
120 PRINT:PRINT "WELL, ";A$;",":PRINT "I AM THINKING OF A"
130 PRINT "BOOK OF THE BIBLE."
140 PRINT:PRINT "YOU HAVE 12 CHANCES":PRINT "TO GUESS WHICH ONE."
150 R=INT(RND(1)*66)+1
160 G=0
170 G=G+1
180 IF G=13 THEN 600
190 PRINT "THIS IS CHANCE #";G;"."
200 RESTORE
210 FOR B=1 TO R
220 READ B$
230 NEXT B
240 INPUT "YOUR GUESS";G$
310 IF G$=B$ THEN 500
320 RESTORE
330 X=0
340 X=X+1
360 IF X>66 THEN PRINT "***SPELLING ERROR***":GOTO 170
363 READ S$
364 IF S$=G$ THEN 370
365 GOTO 340
370 C=ABS(R-X)
380 IF C=1 THEN PRINT "NEXT DOOR NEIGHBORS!"
390 IF C=2 THEN PRINT "YOU'RE EXTREMELY CLOSE!"
400 IF C>2 AND C<5 THEN PRINT "LESS THAN 5 BOOKS AWAY."
410 IF C>4 AND C<10 THEN PRINT "LESS THAN 10 AWAY!"
420 IF C>9 AND C<20 THEN PRINT "COULD BE WORSE..."
430 IF C>19 AND C<30 THEN PRINT "YOU'LL NEVER GET IT!"
440 IF C>29 AND C<40 THEN PRINT "WAY OFF, ";A$;"."
450 IF C>39 AND C<50 THEN PRINT "YOU CAN DO MUCH BETTER."
460 IF C>49 AND C<60 THEN PRINT "NOWHERE NEAR!"
470 IF C>59 AND C<66 THEN PRINT "FAR, FAR AWAY!"
480 GOTO 170
500 PRINT "YOU ARE RIGHT!": PRINT "THAT'S IT!"
510 PRINT: PRINT "CAN YOU DO IT AGAIN?"
520 INPUT "YES OR NO";R$
530 IF R$="YES" THEN GOSUB 700
540 IF R$="NO" THEN 900
550 GOTO 500
560 RETURN
```

```
600 PRINT "TOO BAD, ";A$;"."
610 PRINT "I WAS THINKING OF"
620 PRINT B$;"."
630 PRINT:PRINT "WANT TO TRY AGAIN?"
640 INPUT "YES OR NO"; P$
650 IF P$="YES" THEN 700
660 IF P$="NO" THEN 800
670 GOTO 640
700 PRINT "GOOD! GOOD! GOODY!":PRINT "I JUST LOVE THIS!"
710 FOR T=1 TO 1000: NEXT
720 PRINT "NOW LET ME THINK...":PRINT "WHICH BOOK THIS TIME?"
730 FOR T=1 TO 1500: NEXT
740 PRINT:PRINT "ALRIGHT, I'VE GOT IT!"
750 FOR T=1 TO 200: NEXT
760 GOTO 150
800 PRINT "I'M SORRY, ";A$;",":PRINT "STUDY SOME MORE"
810 PRINT "AND TRY ANOTHER DAY."
820 END
900 PRINT "YOU WERE AN ENJOYABLE": PRINT
910 PRINT "COMPANION. PLEASE TRY": PRINT
920 PRINT "ANOTHER DAY. PEACE!"
930 END
1000 DATA"GENESIS","EXODUS","LEVITICUS","NUMBERS"
1005 DATA"DEUTERONOMY","JOSHUA","JUDGES","RUTH"
1010 DATA"1 SAMUEL","2 SAMUEL","1 KINGS","2 KINGS"
1015 DATA"1 CHRONICLES","2 CHRONICLES","EZRA","NEHEMIAH","ESTHER"
1020 DATA"JOB","PSALMS","PROVERBS","ECCLESIASTES","SONG OF SOLOMON"
1025 DATA"ISAIAH","JEREMIAH","LAMENTATIONS","EZEKIEL"
1030 DATA"DANIEL","HOSEA","JOEL","AMOS","OBADIAH","JONAH"
1035 DATA"MICAH","NAHUM","HABAKKUK","ZEPHANIAH","HAGGAI"
1040 DATA"ZECHARIAH","MALACHI","MATTHEW","MARK","LUKE","JOHN"
1045 DATA"ACTS","ROMANS","1 CORINTHIANS","2 CORINTHIANS"
1050 DATA"GALATIANS","EPHESIANS","PHILIPPIANS","COLOSSIANS"
1055 DATA"1 THESSALONIANS","2 THESSALONIANS","1 TIMOTHY"
1060 DATA"2 TIMOTHY","TITUS","PHILEMON","HEBREWS"
1065 DATA"JAMES","1 PETER","2 PETER","1 JOHN","2 JOHN"
1070 DATA"3 JOHN","JUDE","REVELATION"
```

The program printed on the next page demonstrates how the BASIC "Which Book?" program can be adapted for some special effects on the VIC-20. The resulting game is essentially the same as the one above but has the interesting addition of sound effects.

It is necessary to add the 3K Super Expander cartridge to your VIC-20 in order to LOAD and RUN this version of the program. Line 501 is a PRINT statement beginning with a reverse *F*. This is a music line that can be interpreted only by the Super Expander.

Compare the two versions of this program. You will notice that many things have changed in addition to the introduction of audio. "Crunching" was the first requirement. In order to conserve memory, all unnecessary spaces were eliminated. This makes the program harder to read, but it is a standard practice. You will also note that a few lines have been doubled up with a colon separating what were formerly two lines of code. Many shortcuts that are unique to the VIC-20 were also taken for line spacing.

If your computer is a VIC-20, by all means enter the game from

this program the first time. Using this as an example, you will see many ways the BASIC program can be adapted for any brand of computer. The audio for other computers would be placed in essentially the same spots in the program.

Which Book? (VIC-20 version)

```
10 REM "WHICH BOOK?" (VIC 20 VERSION)
100 PRINT "□□□□□□HELLO, WHAT'S YOUR□
110 INPUT "NAME";A$
120 PRINT "□□□□□□□□WELL, ";A$;",":PRINT "I AM THINKING OF A"
130 PRINT "BOOK OF THE BIBLE."
134 FORT=1TO1700:NEXT
140 PRINT "□□□□YOU HAVE 12 CHANCES   TO GUESS WHICH ONE."
144 FORT=1TO2400:NEXT
150 R=INT(RND(1)*66)+1
160 G=0
170 G=G+1
180 IFG=13THEN600
190 PRINT"□□□□THIS IS CHANCE #";G;".□"
200 RESTORE
210 FORB=1TOR
220 READB$
230 NEXTB
240 INPUT"YOUR GUESS";G$
250 POKE36878,15
255 FORM=1TO50
260 POKE36876,INT(RND(1)*128)+128
265 FORS=1TO10:NEXTS
270 NEXTM
280 POKE36876,0:POKE36878,0
310 IFG$=B$THEN500
320 RESTORE
330 X=0
340 X=X+1
360 IFX>66THENPRINT"□□*****SPELLING ERROR*****■":GOTO170
363 READS$
364 IFS$=G$THEN370
365 GOTO340
370 C=ABS(R-X)
380 IFC=1THENPRINT"□□NEXT DOOR NEIGHBORS!■□"
390 IFC=2THENPRINT"□□YOU'RE EXTREMELY CLOSE■□"
400 IFC>2ANDC<5THENPRINT"□□LESS THAN 5 BOOKS AWAY■□"
410 IFC>4ANDC<10THENPRINT"□□LESS THAN 10 AWAY!■□"
420 IFC>9ANDC<20THENPRINT"□□COULD BE WORSE...■□"
430 IFC>19ANDC<30THENPRINT"□□YOU'LL NEVER GET IT!■□"
440 IFC>29ANDC<40THENPRINT"□□WAY OFF, ";A$"■□"
450 IFC>39ANDC<50THENPRINT"□□YOU CAN DO MUCH BETTER■□"
460 IFC>49ANDC<60THENPRINT"□□NOWHERE NEAR!■□"
470 IFC>59ANDC<66THENPRINT"□□FAR, FAR AWAY!■□"
480 GOTO170
500 PRINT"□□□YOU ARE RIGHT!":PRINT"□□□THAT'S IT!"
501 PRINT"■V9S202T2003CEGRET6G
510 PRINT"□□□□CAN YOU DO IT AGAIN?□"
520 INPUT "YES OR NO";R$
530 IFR$="YES"THENGOSUB700
540 IFR$="NO"THEN900
```

```
560 RETURN
600 PRINT"TOO BAD, ";A$;"."
610 PRINT"I WAS THINKING OF"
620 PRINTB$;"."
621 POKE36878,15
622 FORS=1TO5
623 POKE36875,183
624 FORQ=1TO150:NEXTQ
625 POKE36875,0
626 FORQ=1TO150:NEXTQ
627 NEXTS
628 POKE36878,0
630 PRINT"WANT TO TRY AGAIN?"
640 INPUT"YES OR NO"; P$
650 IFP$="YES"THEN700
660 IFP$="NO"THEN800
670 GOTO640
700 PRINT "GOOD! GOOD! GOODY!":PRINT "I JUST LOVE THIS!"
702 POKE36878,15
704 FORM=1TO75
706 POKE36874,INT(RND(1)*100)+100
708 FORS=1TO25:NEXTS
710 NEXTM
715 POKE36874,0:POKE36878,0
720 PRINT"NOW LET ME THINK. . ."
724 FORT=1TO800:NEXT
726 PRINT"WHICH BOOK THIS TIME?"
730 FORT=1TO2000:NEXT
740 PRINT "ALRIGHT, I'VE GOT IT!"
742 POKE36878,15
744 FORM=1TO35
746 POKE36875,INT(RND(1)*128)+128
748 FORS=1TO70:NEXTS
750 NEXTM
753 POKE36875,0:POKE36878,0
755 PRINT""
760 GOTO150
800 PRINT"I'M SORRY, ";A$;",":PRINT "STUDY SOME MORE"
810 PRINT"AND TRY ANOTHER DAY."
815 PRINT"
820 END
900 PRINT "YOU WERE AN ENJOYABLE"
910 PRINT "COMPANION. PLEASE TRY"
920 PRINT "ANOTHER DAY. PEACE!"
930 END
1000 DATA"GENESIS","EXODUS","LEVITICUS","NUMBERS"
1005 DATA"DEUTERONOMY","JOSHUA","JUDGES","RUTH"
1010 DATA"1 SAMUEL","2 SAMUEL","1 KINGS","2 KINGS"
1015 DATA"1 CHRONICLES","2 CHRONICLES","EZRA","NEHEMIAH","ESTHER"
1020 DATA"JOB","PSALMS","PROVERBS","ECCLESIASTES","SONG OF SOLOMON"
1025 DATA"ISAIAH","JEREMIAH","LAMENTATIONS","EZEKIEL"
1030 DATA"DANIEL","HOSEA","JOEL","AMOS","OBADIAH","JONAH"
1035 DATA"MICAH","NAHUM","HABAKKUK","ZEPHANIAH","HAGGAI"
1040 DATA"ZECHARIAH","MALACHI","MATTHEW","MARK","LUKE","JOHN"
1045 DATA"ACTS","ROMANS","1 CORINTHIANS","2 CORINTHIANS"
1050 DATA"GALATIANS","EPHESIANS","PHILIPPIANS","COLOSSIANS"
1055 DATA"1 THESSALONIANS","2 THESSALONIANS","1 TIMOTHY"
1060 DATA"2 TIMOTHY","TITUS","PHILEMON","HEBREWS"
1065 DATA"JAMES","1 PETER","2 PETER","1 JOHN","2 JOHN"
1070 DATA"3 JOHN","JUDE","REVELATION"
```

This version of the game was tested, exactly as printed, on a group of fourteen eighth-grade boys and girls. The results were impressive. The young people stayed with it for forty-five minutes and exhibited increasing interest up to the final moment.

A week before, the class had reviewed the structure of the Bible and surveyed the various styles of the book groupings. When they entered the classroom on the day of the game, the computer was already set up. The screen displayed "READY" and a flashing cursor. Immediate curiosity and high anticipation were clearly evident.

As they settled down, I began a quick review of what they had learned the week before, while each student held a Bible opened to the table of contents. I made a point of pronouncing the name of each book as we came to it. In less than ten minutes we were ready to turn to the computer.

I typed in "RUN," and a simple three-line program scrolled "HELLO, CLASS!" across the screen. Most of the class laughed and spoke to the computer. I then let them see me LOAD the game from a cassette. Their curiosity was very high.

I had decided that it would be best, under the circumstances, for me to work the keyboard while letting them call the shots. I asked for a volunteer. Everyone nervously avoided being first. They began to point to a guinea pig who responded to the challenge with delight. He called out "Psalms," an easy, safe choice. The first clue was rather encouraging and in a matter of six tries, he found the answer.

Now that they had seen how it worked, everyone was eager for a turn. I selected the girl with the most frantically waving hand and asked her how she spelled her name. Immediately, I saw a feature of the game that I had not expected. This was a new group—new to me and new to each other. By putting each individual's name on the screen in the opening A$ gimmick, we all got a little better acquainted with each other! Each time the name popped up during the RUN we were reminded of the name of the person calling out the guesses. For me, this was a delightful discovery. There are all kinds of fringe benefits from playing Bible games with a computer!

After eight or nine games I began to see that twelve chances were about four too many. Most of them could guess the correct book within about seven guesses. One lucked out in three. All of them got it inside of twelve. No one ever lost a game, which was a pity. I had put in a very discouraging "you lose" sound effect, which the class never got to hear.

They were working with their Bibles open at first. After each individual who wanted a turn had played the game, I then made them close their Bibles and work entirely from memory. "This time," I thought, "they'll get to hear my wonderful raspberries!"

To keep from embarrassing any individual who might not be able to keep up with the game when working from memory alone, I went around the room, asking each person to suggest one book from the Bible in turn. A tremendous sense of teamwork against the computer developed as they coached each other in possible next guesses.

We played the game group-style four times, starting in different places among the seats each time, and the group won them all. On the last game it took all twelve chances, and when the correct answer signal flashed up, the young people cheered as though they were at a ball game.

Then, just for fun, I instructed the computer to play another round with the same A$ INPUT so they could see the "GOOD! GOOD! GOODY! I JUST LOVE THIS!" routine. I had added a little electronic bubbling as audio. They were transfixed. Out of the silence, someone commented, "I thought computers didn't have any feelings. . . ."

In short, the game was a success in every way. Time passed unusually quickly and we learned a lot more about each other and the order of the books of the Bible than anyone present consciously noticed.

12

DAVID AND GOLIATH

The famous battle between young David and the Philistine giant Goliath is recorded in 1 Samuel 17. The huge man from Gath, wearing many pounds of bronze armor, was a tremendous threat to the Israelites. Then David, with his slingshot, threw a stone that hit him on the forehead and fractured his skull.

In this game, the computer plays David and you play Goliath. The field of battle is an eight-by-eight grid. David's position is marked by the letter *D*, and Goliath's by the letter *G*.

David is programmed to be aggressive. The Bible tells us that

the Lord. The computer exemplifies that trust by never running away. David will always come after you, no matter where you may dodge. He will gradually outwit you and come within easy slingshot range. He *always* wins.

The object of the game is to dodge as many of his stones as possible. Each successful round doubles your score. If you stay in the game too long, David will hit you and all points will be lost. The important thing is to avoid him as long as possible and then "chicken out" before you think he will connect.

To accomplish this, you have a choice of four moves for Goliath. He can move up the grid (press *U*, and return). He can move to the right (press *R*, and return). He can move down the screen (press *D*, and return). He can move to the left (press *L*, and return). A fifth option is to stay where you are (press *S*, and return). You also have the important advantage of three "giant steps," which allow you to move twice as far with each turn (press *G*, return; then the direction you want; then return).

The way to "chicken out" is deliberately to run off the grid. If you are successful, your score will appear on the screen. The version which we have adapted for the VIC-20 includes a way to let the high scorer place three initials above the instructions. You may be able to adapt this for other brands of computers. Scores in excess of one thousand are possible, so don't give up too soon!

Line 200 places Goliath in a random position on the first column at the start of the game. Line 210 lets David materialize in surprising places on the last column. This feature makes each round a little different from the one before.

Line 251 limits the number of "giant steps" to three. You can change this number to suit your own taste, but three seems about right. The size of the special movement is determined by line 252. You can let the variable *M* equal three or four if you want to see how it works. Player movement is controlled by line 290. The computer's movement of David is directed by lines 340 through 360.

A particularly nice detail in this game is the fact that your odds of getting bonked on the noggin by a stone from David's slingshot are determined by logic. If you are out of range there is no chance of being struck. If you are within two spaces in any direction of each other, lines 430 through 470 combined with line 510 create a 10 percent chance that you will be hit. If the gap closes to one space, the odds go up to 50 percent. You can see how this works in lines 380 through 420, with line 500 casting the lots. Line 375 wipes out

You can change the size of the grid by adjusting the numbers in lines 310, 900, and 920. For fun, try some rectangular grids.

The use of DATA in this program is unique. It saves many boring IF THEN statements. Line 30 reads it and sets up a command string for the five general orders: *U, D, R, L, S*. The numbers after each letter tell the computer where to go by adding one, subtracting one, or doing nothing. Hence, *S* (Stay) is followed by double zeros.

Take the time to dress up the opening instructions in ways suited to your personal computer. Let your imagination work on it! There are many opportunities throughout this game for the introduction of audio and graphics. Some sort of disappointing "falling" sound would make Goliath's defeat especially impressive.

David and Goliath

```
1 REM  "DAVID AND GOLIATH"
10 DIM C$(5),MX(5),MY(5):SC=1:M=1
20 FOR L=1 TO 5
30 READ C$(L),MX(L),MY(L)
40 NEXT L
100 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
110 PRINT "* DAVID AND GOLIATH *"
120 PRINT:PRINT "THE COMPUTER IS DAVID, YOU ARE GOLIATH."
130 PRINT "STAY OUT OF DAVID'S STONE THROWING RANGE."
140 PRINT "MOVE GOLIATH UP, DOWN, LEFT, OR RIGHT.  YOU MAY ALSO STOP."
150 PRINT:PRINT "USE U,D,R,L,S AS COMMANDS."
155 PRINT:PRINT "TYPE 'G' TO MAKE GIANT STEPS."
160 PRINT:PRINT "MOVING OFF OF THE GRID ENDS THE GAME AND REGISTERS YOUR SCORE."
170 PRINT:PRINT "PRESS RETURN TO START"
180 INPUT A$
190 PRINT "□":REM REPLACE AS ABOVE.
200 GX=1:GY=INT(RND(1)*8)+1
210 DX=8:DY=INT(RND(1)*8)+1
220 GOSUB 900
230 PRINT:PRINT "YOUR MOVE";
240 INPUT A$
250 FOR L=1 TO 5
251 IF A$="G" AND GS=3 THEN PRINT "NO MORE GIANT STEPS - YOU ARE TIRED":GOTO 230
252 IF A$="G" THEN M=2:GS=GS+1:PRINT "DIRECTION";:GOTO240
260 IF A$=C$(L) THEN 290
270 NEXT L
280 PRINT "WHAT ?":GOTO 230
290 GX=GX+MX(L)*M:GY=GY+MY(L)*M
295 DN=0:M=1
300 GOSUB 900
310 IF GX<1 OR GX>8 OR GY<1 OR GY>8 THEN 600
320 FOR T=1 TO 250:NEXT T
330 PRINT:PRINT "COMPUTER'S MOVE:"
340 XD=SGN(GX-DX)
350 YD=SGN(GY-DY)
360 DX=DX+XD:DY=DY+YD
370 GOSUB 900:PRINT
375 IF DX=GX AND DY=GY THEN 700
380 FOR X=-1 TO 1
390 FOR Y=-1 TO 1
400 CX=GX+X:CY=GY+Y
410 IF DX=CX AND DY=CY THEN DN=1:PRINT "* DANGER !":GOTO 500
420 NEXT Y:NEXT X
430 FOR X=-2 TO 2
440 FOR Y=-2 TO 2
450 CX=GX+X:CY=GY+Y
460 IF DX=CX AND DY=CY THEN DN=2:PRINT "* WATCH OUT":GOTO 500
470 NEXT Y:NEXT X
480 PRINT "* OUT OF RANGE."
490 GOTO 530
```

```
500 IF DN=1 THEN RN=INT(RND(1)*2)+1
510 IF DN=2 THEN RN=INT(RND(1)*10)+1
520 IF RN=1 THEN 700
530 SD=SD+1
540 SC=SC*2
550 PRINT "STONES DODGED:";SD
560 PRINT "CURRENT SCORE:";SC
570 GOTO 230
600 PRINT:PRINT "  ** CHICKEN ! **"
610 PRINT:PRINT "YOU ESCAPED FROM DAVID WITH A SCORE OF";SC;"."
620 PRINT:PRINT "YOU DODGED";SD;"STONES TO GET THAT SCORE."
630 GOTO 800
700 PRINT:PRINT "* DAVID GOT YOU !"
710 PRINT:PRINT "AT A RANGE OF";DN;"UNITS."
720 IF DN=2 THEN PRINT "A LUCKY SHOT.":GOTO 750
730 IF DN=1 THEN PRINT "AN EASY HIT.":GOTO 750
740 PRINT "AN OBVIOUS KILL."
750 PRINT:PRINT "UNFORTUNATELY, YOU LOST YOUR SCORE OF";SC;
760 PRINT "AS WELL AS YOUR LIFE."
800 PRINT:PRINT "WOULD YOU LIKE TO TRY AGAIN ?"
810 INPUT A$
820 IF LEFT$(A$,1)="N" THEN PRINT "BYE.":END
830 RUN
899 END
900 FOR Y=1 TO 8
910 PRINT TAB(6);
920 FOR X=1 TO 8
930 IF X=GX AND Y=GY THEN PRINT "G";:GOTO 970
940 IF X=DX AND Y=DY THEN PRINT "D";:GOTO 970
950 PRINT ".";
970 NEXT X:PRINT
980 NEXT Y
990 RETURN
995 DATA U,0,-1,D,0,1,R,1,0,L,-1,0,S,0,0
```

As an example of how this program can be expanded and "dressed up" on a specific computer, a version prepared especially for the Commodore VIC-20 is printed on pages 67–68. This printout has some peculiarities that need to be explained. Our challenge was to see how much could be done within the limits of an unexpanded VIC-20. As a result, many neat, eye-saving spaces had to be eliminated. When you type this program yourself, you may omit all of the spaces in the code lines. A few of the lines also "wrap around" (run to the following line). Be careful; patiently work down each line.

Every unusual character in the printout has a simple explanation. Follow along. Line 9 POKES the entire screen dark blue. The thing inside the quotation marks is a reverse *E*, obtained by pressing CTRL (Control key) and the number 2. This makes white characters appear on a dark blue field.

Line 11 sets up a preliminary high score. Line 99 resets the score to zero for each new round without changing the high score. The snaggle-toothed appearance of lines 120 through 160 results from the spacing needed for the text on the VIC-20 screen. For

example, the space in line 160 between "GRID" and "ENDS" is automatic, while the five spaces between "AND" and "REGISTERS" prevent the division of a word. This line is also the first in the program to wrap around in the printout.

A truly high-tech approach to instructions begins the game. Line 162 POKES the vertical position of the screen down out of sight while it is being set up. Line 161 then scrolls it into view. Lines 181 through 184 flash the initials of the high scorer at the top of the screen while the computer waits for the GET A$ in line 180. The reverse *S* in line 181 is HOME. The reverse *R* is obtained by pressing CTRL and RVS ON. Incidentally, using a GET eliminates the need of pressing RETURN after each move.

The body of the game has already been explained. New material for the VIC-20 appears again at line 800. Inside the quotation marks are three spaces, fifteen pressings of @ with the COMMODORE key held down. This makes a line that will help the tops of the letters in "Play again?" stand out. The *R* before the words is another REVERSE ON. The asterisks are decoration.

Line 900 follows the quotation mark with HOME, five spaces, one COMMODORE plus *D*, eight COMMODORE plus *I*, and one COMMODORE plus *F*. Line 910 has five spaces, REVERSE ON, COMMODORE plus *K*, and REVERSE OFF. Line 970 has only the vertical bar produced by COMMODORE plus *K*. Line 985 uses five spaces, COMMODORE plus *C*, REVERSE ON, eight COMMODORE plus *I*, REVERSE OFF, and COMMODORE plus *V*. Line 986 has twenty-one spaces after the quotation mark in the first part. The second part has HOME followed by nine cursor down marks. Although this may look terribly complicated, it isn't. It produces a beautiful grid that is well worth the trouble.

Three sound effects are contained in the subroutines in line 1000 through 1320.

If you have a memory-expanding cartridge you can add many more details of your own. Remember that this program flirts with the limit of memory as it is printed here. If you drop the spaces you will pick up another hundred or two BYTES.

David and Goliath (VIC-20 version)

```
0 REM "DAVID AND GOLIATH" (VIC 20 VERSION)
9 POKE36879,110:PRINT "■"
10 DIM C$(5),MX(5),MY(5):SC=1:M=1
11 HS=4   :IN$="CKW"
12 VL=36878:S1=36875:S2=36876
20 FOR L=1 TO 5
30 READ C$(L),MX(L),MY(L)
40 NEXT L
99 GS=0:SC=1:SD=0:POKE 36865,125
100 PRINT "□"
110 PRINT "* DAVID AND GOLIATH *"
120 PRINT:PRINT "THE COMPUTER IS DAVID,YOU ARE GOLIATH."
130 PRINT "STAY OUT OF DAVID'S   STONE THROWING RANGE."
140 PRINT "MOVE GOLIATH UP, DOWN, LEFT, OR RIGHT.  YOU MAY ALSO STOP."
150 PRINT:PRINT "USE U,D,R,L,S AS      COMMANDS."
155 PRINT:PRINT "TYPE 'G' TO MAKE GIANT STEPS."
160 PRINT:PRINT "MOVING OFF OF THE GRIDENDS THE GAME AND     REGISTERS YOUR SCOR
E."
161 FOR L=125 TO 25 STEP -.5
162 POKE 36865,L:NEXT
170 PRINT:PRINT " PRESS SPACE TO START":GOSUB 1100
180 GET A$:IF A$<>"" THEN 185
181 PRINT "■ ■HIGH SCORE:";HS;IN$
182 FOR T=1TO200:NEXT:PRINT "■ HIGH SCORE:";HS;IN$
183 FOR T=1 TO 200:NEXT
184 GOTO 180
185 GOSUB 1000:FOR L=25 TO 125:POKE 36865,L:NEXT:PRINT "□":POKE 36865,25
200 GX=1:GY=INT(RND(1)*6)+1
210 DX=8:DY=INT(RND(1)*6)+1
220 GOSUB 900
230 PRINT:PRINT "YOUR MOVE"
240 GET A$:IF A$="" THEN 240
241 GOSUB 1000
250 FOR L=1 TO 5
251 IF A$="G" AND GS=3 THEN PRINT "NO MORE GIANT STEPS":GOTO 230
252 IF A$="G" THEN M=2:GS=GS+1:PRINT "DIRECTION?" :GOTO 240
260 IF A$=C$(L) THEN 290
270 NEXT L
280 PRINT "WHAT ?":GOTO 230
290 GX=GX+MX(L)*M:GY=GY+MY(L)*M
295 DN=0:M=1
300 GOSUB 900
310 IF GX<1 OR GX>8 OR GY<1 OR GY>8 THEN 600
320 FOR T=1 TO 250:NEXT T
325 POKE VL,15:POKE S1,150:FOR T=1 TO 50:NEXT:POKE S1,0:POKE VL,0
330 PRINT:PRINT "COMPUTER'S MOVE:"
340 XD=SGN(GX-DX)
350 YD=SGN(GY-DY)
360 DX=DX+XD:DY=DY+YD
370 GOSUB 900:PRINT
375 IF DX=GX AND DY=GY THEN 700
```

```
380 FOR X=-1 TO 1
390 FOR Y=-1 TO 1
400 CX=GX+X:CY=GY+Y
410 IF DX=CX AND DY=CY THEN DN=1:PRINT "■* DANGER !":GOSUB 1100:GOTO 500
420 NEXT Y:NEXT X
430 FOR X=-2 TO 2
440 FOR Y=-2 TO 2
450 CX=GX+X:CY=GY+Y
460 IF DX=CX AND DY=CY THEN DN=2:PRINT "■* WATCH OUT":GOSUB 1300: GOTO 500
470 NEXT Y:NEXT X
480 PRINT "* OUT OF RANGE."
490 GOTO 530
500 IF DN=1 THEN RN=INT(RND(1)*2)+1
510 IF DN=2 THEN RN=INT(RND(1)*10)+1
520 IF RN=1 THEN 700
530 SD=SD+1
540 SC=SC*2
550 PRINT "STONES DODGED:";SD
560 PRINT "CURRENT SCORE:";SC
570 GOTO 230
600 PRINT:PRINT "]   ** CHICKEN ! **":GOSUB 1300
610 PRINT:PRINT "YOU ESCAPED FROM DAVID WITH A SCORE OF";SC;"."
619 PRINT:PRINT "YOU DODGED";SD;"STONES TO GET THAT SCORE."
620 IF SC<=HS THEN 800
621 GOSUB 1100:PRINT "■YOU HAVE A NEW HIGH   SCORE !"
622 PRINT "■ENTER YOUR INITIALS.":PRINT ">>>";
623 IN$="":HS=SC:FOR L=1 TO 3
624 GET A$:IF A$="" THEN 624
625 GOSUB 1000:IN$=IN$+A$:PRINT A$;:NEXT:PRINT
626 PRINT "THANK YOU.":FOR T=1 TO 500:NEXT
630 GOTO 99
700 PRINT:PRINT "]* DAVID GOT YOU !":GOSUB 1200
710 PRINT:PRINT "AT A RANGE OF";DN;"UNITS."
720 IF DN=2 THEN PRINT "A LUCKY SHOT.":GOTO 750
730 IF DN=1 THEN PRINT "AN EASY HIT.":GOTO 750
740 PRINT "AN OBVIOUS KILL."
750 PRINT:PRINT "UNFORTUNATELY, YOU LOST YOUR SCORE OF";SC;"AS WELL AS YOUR LIFE
."
800 PRINT "     ________________":PRINT "    ■*PLAY AGAIN ? *"
810 GET A$:IF A$="" THEN 810
820 IF LEFT$(A$,1)="N" THEN PRINT "BYE.":END
830 GOSUB 1000:GOTO 99
899 END
900 PRINT "■       ■■■■■■■■■■"
905 FOR Y=1 TO 8
910 PRINT "        ■■";
920 FOR X=1 TO 8
930 IF X=GX AND Y=GY THEN PRINT "G";:GOTO 970
940 IF X=DX AND Y=DY THEN PRINT "D";:GOTO 970
950 PRINT "+";
970 NEXT X:PRINT "■"
980 NEXT Y
985 PRINT "         ■■■■■■■■■■"
986 FOR L=1 TO 12:PRINT "                              ":NEXT:PRINT "■■■■■■■■■■"
990 RETURN
1000 POKE VL,15:POKE S2,230:FOR T=1 TO 75:NEXT:POKE S2,0:POKE VL,0:RETURN
1100 POKE VL,15:POKE S2,200:POKE S1,200:FOR T=1 TO 75:NEXT
1110 FOR L=15 TO 0 STEP-1:POKE VL,L:FOR T=1 TO 10:NEXT:NEXT:POKE S1,0:POKE S2,0
1120 POKE VL,15:POKE S2,210:FOR T=1 TO 50:NEXT:POKE S2,0:FOR T=1 TO 40:NEXT:POKE
 S2,210
1130 FOR T=1 TO 50:NEXT:POKE S2,0:POKE VL,0:RETURN
1200 POKEVL,15:FORL=255TO128STEP-1:POKES1,L:POKES2,L:NEXT:POKES1,0:POKES2,0
1210 POKE36877,255:FORL=15TO 0STEP-1:POKEVL,L:FORT=1TO50:NEXT:NEXT:POKE36877,0:R
ETURN
1300 POKEVL,15:POKES2,210:FORT=1TO120:NEXT:POKES2,0:FORT=1TO50:NEXT:POKES2,210
1310 FORT=1TO100:NEXT:POKES2,0:FORT=1TO50:POKES1,200:FORT=1TO400:NEXT:POKES1,0
1320 RETURN
1999 DATA U,0,-1,D,0,1,R,1,0,L,-1,0,S,0,0
```

13

MEMORY VERSES

This program uses very little memory and can be expanded to very large proportions simply by adding more DATA. In its printed form, the game will teach the player eleven verses of scripture. And *teach* is not a careless choice of words. Above all else, this game is a teacher.

Let's see how it works. The game starts by giving the player three choices of difficulty. Let's assume that the first-time player chooses the easiest, category 1. That means that the computer will randomly select one of the verses stored in DATA statements and

computer will state which book it is from and the chapter number. All the player is asked to submit is a number for the verse. If the suggested number is correct, the random generator in line 310 will select another verse from the DATA and continue the game. If the player's answer was incorrect, the screen will promptly say so and reveal the true answer. In that moment, education takes place.

In difficulty level 2, the computer announces only the name of the book that the verse is taken from and asks the player for chapter and verse. Again, wrong answers are followed by a peek at the right answer.

Level 3 is playing cold turkey. The computer quotes the verse and then backs out of it, asking the player for book, chapter, and verse. As before, correct answers are revealed whenever a wrong answer is given.

With the eleven verses given in the printed program, most people can get a nearly perfect score after one time through each of the three difficulty levels. Each one of the verses is asked in every game, but in constantly varying order.

This bare-bones program presents some wonderful opportunities for creative decoration, but its greatest asset is the size of the DATA bank it can hold on even a limited-memory computer. It can quote an enormous number of verses on less than 5K.

Take a look at one of the DATA statements and absorb the special style of format. The examples are from the King James Version, but you can use any version you wish. The verse, or a major portion of it, is written first, enclosed in quotation marks, and followed by a comma. Then the book is given (with no space after the comma), followed by a second comma. Again without leaving any space, the chapter number, comma, and verse number are entered. Following this procedure, you can feed in as many verses as you wish.

Once you have finished entering DATA, count the number of DATA lines and enter that number as the large figure in lines 300 and 310. As in all such cases, accuracy in entering the DATA is important.

If others will be playing this game, be sure to tell them whether or not you have spelled Psalms in the plural, and so on. Otherwise, the players may protest that they gave the correct information but the computer called a strike.

Unfortunately, *the traditional colon between chapter and verse* (John 3:16) *can't be used*. The colon is too powerful an input

Tell your players to substitute a comma (John 3,16) *when they attempt levels two and three.*

This game is a good one to play alone. If you have a list of verses you want to be able to call at the snap of a finger, put them into this program and spend some time with them. You will be surprised at what an excellent tutor the computer can be. If you stick with it, you can be sure you will learn them all. Start by giving verse only and work up to the more difficult levels.

Memory Verses

```
10 REM  "MEMORY VERSES"
100 PRINT "□":REM  REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
105 DIM F(11)
110 PRINT "  MEMORY VERSES."
120 PRINT:PRINT "I WILL QUOTE BIBLE"
130 PRINT "VERSES, AND ASK YOU"
140 PRINT "FOR THE SOURCE."
150 PRINT "CHOSE DIFFICULTY-"
160 PRINT "     -CLUES-"
180 PRINT:PRINT "(1) BOOK AND CHAPTER"
190 PRINT "(2) BOOK ONLY"
200 PRINT "(3) NO CLUES"
210 PRINT:PRINT "FEWER CLUES GIVE A"
220 PRINT "HIGHER SCORE."
230 INPUT "CHOICE>";D
240 IF D<1 OR D>3 THEN PRINT "#'S 1-3 ONLY":GOTO 160
300 FOR CY=1 TO 11
310 RI=INT(RND(1)*11)+1
320 IF F(RI)=1 THEN 310
330 RESTORE
340 FOR I=1 TO RI
350 READ Q$,B$,C,V
360 NEXT I
370 F(RI)=1
380 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
400 PRINT "QUESTION #";CY
410 PRINT:PRINT Q$
420 PRINT
430 IF D=1 THEN PRINT B$;C;":?":I$=B$:I=C
440 IF D=2 THEN PRINT B$;"-?:?":I$=B$
450 IF D=3 THEN PRINT "?-?:?"
460 PRINT
470 IF D=3 THEN INPUT "BOOK";I$
480 IF D=2 OR D=3 THEN INPUT "CHAPTER";I
490 INPUT"VERSE";X
500 IF I$=B$ AND I=C AND X=V THEN 600
510 PRINT "□":W=W+1:REM  KEEP FORMULA IF REPLACING CLEAR SCREEN COMMAND
520 PRINT "SORRY, THAT'S WRONG."
530 PRINT:PRINT "THE CORRECT ANSWER IS"
540 PRINT:PRINT B$;"-";C;":";V
550 FOR I=1 TO 2000:NEXT I
560 GOTO 700
600 PRINT:PRINT "CORRECT !"
610 FOR I=1 TO 500:NEXT I
620 GOTO 700
700 NEXT CY
705 CY=CY-1
710 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
720 PRINT "GAME OVER."
730 PRINT:PRINT "YOU GOT";CY-W;" OUT OF"
```

```
750 I=(CY-W)*D*10
760 PRINT:PRINT "YOUR SCORE IS";I
770 IF W=0 THEN PRINT "EXELLENT !"
780 IF W<3 THEN PRINT "GOOD."
790 IF W>2 THEN PRINT "YOU NEED PRACTICE."
800 PRINT:INPUT "TRY AGAIN";I$
810 IF I$="NO" OR I$="N" THEN END
820 RUN
1000 DATA "MAKE A JOYFUL NOISE UNTO THE LORD, ALL YE LANDS.",PSALMS,100,1
1010 DATA "FOR GOD SO LOVED THE WORLD...",JOHN,3,16
1020 DATA "BLESSED ARE THE MEEK, FOR THEY SHALL INHERIT THE EARTH.",MATTHEW,5,5
1030 DATA "IN THE BEGINNING GOD CREATED THE HEAVEN AND THE EARTH.",GENESIS,1,1
1040 DATA "JESUS INCREASED IN WISDOM AND STATURE, AND IN FAVOR...",LUKE,2,52
1050 DATA "TO EVERYTHING THERE IS A SEASON AND A TIME...",ECCLESIASTES,3,1
1060 DATA "THOUGH I SPEAK WITH TONGUES OF MEN AND OF ANGELS...","1 CORINTHIANS",
13,1
1070 DATA "THE LORD IS MY SHEPHERD, I SHALL NOT WANT",PSALMS,23,1
1080 DATA "WHAT DOTH THE LORD REQUIRE OF THEE, BUT TO DO JUSTLY...",MICAH,6,8
1090 DATA "IN THE BEGINNING WAS THE WORD, AND THE WORD WAS WITH GOD...",JOHN,1,1
1100 DATA "AS MANY AS RECEIVED HIM...BECOME THE SONS OF GOD...",JOHN,1,12
```

What follows is a version of "Memory Verses" adapted for the TRS-80, Model II. Note the use of INKEY$ in line 550 and RANDOM in 103. The random number formula in line 310 is typical of TRS-80 BASIC.

Memory Verses (TRS-80 version)

```
100 REM MEMORY VERSES; TRS-80 MOD. ][ VERSION
101 CLS
103 RANDOM
105 DIM F(11)
110 PRINTTAB(30)"MEMORY VERSES"
115 PRINT:PRINT
120 PRINT "I WILL QUOTE BIBLE VERSES, AND THEN ASK YOU FOR THE SOURCE."
130 PRINT
140 PRINT "CHOOSE THE DIFFICULTY LEVEL THAT SUITS YOU."
160 PRINT
170 PRINTTAB(15);"     -CLUES-"
171 PRINT
172 PRINTTAB(10);"(1) BOOK AND CHAPTER GIVEN"
173 PRINT
174 PRINTTAB(10);"(2) BOOK ONLY"
175 PRINT
176 PRINTTAB(10);"(3) NO CLUES AT ALL"
177 PRINT
178 PRINT "HINT: FEWER CLUES WILL INCREASE A QUESTION'S POINT VALUE."
179 PRINT
180 INPUT" CHOICE >";D
190 IF D<1 OR D>3 THEN PRINTTAB(10);"* USE NUMBERS 1-3 ONLY.":GOTO 180
300 FOR CY=1 TO 11
310 RI = INT(RND(11))
320 IF F(RI)=1 THEN 310
330 RESTORE
340 FOR I=1 TO RI
350 READ Q$,B$,C,V
360 NEXT I
370 F(RI)=1
380 CLS
400 PRINTTAB(10);"QUESTION NUMBER ";CY
410 PRINT:PRINT Q$
420 PRINT
430 IF D=1 THEN PRINT B$;C;":?":I$=B$:I=C
440 IF D=2 THEN PRINT B$;"-?:?":I$=B$
450 IF D=3 THEN PRINT "?-?:?"
460 PRINT
470 IF D=3 THEN INPUT "BOOK    >";I$
480 IF D=2 OR D=3 THEN INPUT "CHAPTER >";I
490 INPUT "VERSE   >";X
500 IF I$=B$ AND I=C AND X=V THEN 600
510 CLS:W=W+1
520 PRINT "SORRY, THAT IS WRONG. THE CORRECT ANSWER IS:"
530 PRINT
540 PRINT B$;"-";C;":";V
545 PRINT:PRINT "> PRESS A KEY TO CONTINUE."
550 IF INKEY$="" THEN 550
560 GOTO 700
600 PRINT:PRINTTAB(10);" * CORRECT !"
```

```
620 CLS
700 NEXT CY
705 CY=CY-1
710 CLS
720 PRINTTAB(30);"* GAME OVER *"
730 PRINT:PRINT "YOU ANSWERED ";CY-W;" OUT OF ";CY;" CORRECTLY."
750 I=(CY-W)*D*10
760 PRINT:PRINT "YOUR SCORE IS ";I
770 IF W=0 THEN PRINT "EXCELLENT WORK !"
780 IF W<3 THEN PRINT "GOOD WORK."
790 IF W>2 THEN PRINT "YOU NEED TO STUDY."
800 PRINT
810 INPUT "WOULD YOU LIKE TO TRY AGAIN >";I$
820 IF LEFT$(I$,1)="N" THEN PRINT "OKAY - BYE.":END
830 RUN
1000 DATA "MAKE A JOYFUL NOISE UNTO THE LORD, ALL YE LANDS.",PSALMS,100,1
1010 DATA "FOR GOD SO LOVED THE WORLD...",JOHN,3,16
1020 DATA "BLESSED ARE THE MEEK, FOR THEY SHALL INHERIT THE EARTH.",MATTHEW,5,5
1030 DATA "IN THE BEGINNING GOD CREATED THE HEAVEN AND THE EARTH.",GENESIS,1,1
1040 DATA "JESUS INCREASED IN WISDOM AND IN STATURE, AND IN FAVOR WITH GOD AND
MAN.",LUKE,2,52
1050 DATA "TO EVERYTHING THERE IS A SEASON AND A TIME. . .",ECCLESIASTES,3,1
1060 DATA "THOUGH I SPEAK WITH TONGUES OF MEN AND OF ANGELS. . .",1 CORINTHIANS,
13,1
1070 DATA "THE LORD IS MY SHEPHERD, I SHALL NOT WANT",PSALMS,23,1
1080 DATA "WHAT DOTH THE LORD REQUIRE OF THEE, BUT TO DO JUSTLY. . .",MICAH,6,8
1090 DATA "IN THE BEGINNING WAS THE WORD, AND THE WORD WAS WITH GOD. . .",JOHN,1
,1
1100 DATA "AS MANY AS RECEIVED HIM, TO THEM GAVE HE POWER TO BECOME THE SONS OF
GOD. . .",JOHN,1,12
```

14

TWENTY QUESTIONS

Here is a very simple program that performs like a teacher's dream! Basically, it is a list of twenty biblical personalities, combined with a short statement of something that made each one famous. A quick glance at the DATA beginning at line 1000 will tell you all there is to know. But remember, the player can't see the DATA!

What makes this quiz unique is the fact that it is never presented the same way twice. Line 110 selects a description at random. This means that the order of the twenty questions is always a new arrangement. Line 170 puts a "flag" on each DATA statement as it is used

line 110 that carries such a flag. This prevents repetition and ensures that all twenty questions get asked. You may want to add a device like this to other games, such as "Which Book?"

But that is only the beginning! Not only is the order of questioning constantly varied, but so is the assortment of three wrong answers that accompany each right answer. In other words, the multiple choices themselves are never the same. These are selected by line 520 in cooperation with the loop started in line 500.

Are you ready for this? The very placement of the correct answer in the block of four choices is also stirred from one round to the next by an additional random generator. You will find it in line 650.

What you have, then, is a series of twenty questions that are presented in different order each time around, with different sets of multiple-choice answers each time around, and with the correct answer resting in a different slot each time around. There is no way a "memorizer" can defeat this test. Either the correct answer is known or it is not.

To make it useful as a teaching aid, the correct answer is always displayed on the screen after the player makes a mistake. Lines 720 and 730 take care of this.

You may change the DATA any way you like. You could let it pair up famous Bible couples or biblical books and themes. You can shorten or lengthen the list. If you do, be sure to change the numbers in lines 90, 100, 110, 220, and 520.

The way the program is listed below, there are only three degrees of proficiency that are recognized by the computer—low, medium, and high. These are sorted out in lines 240, 250, and 260. You can easily fine-tune these responses by changing the numbers and adding more gradients to the comments.

There is no time-delay loop after the correct response in line 710. This is intentional. The computer is kept busy enough after returning for a new reading of the DATA to keep the word and its eye-catching asterisks on the screen for about a second. This flash seems to be enough to satisfy most players and keeps the game moving along. One new player, a woman in her seventies, claimed she could not see this notice on the screen. After she played the game several times, her problem disappeared, and she could see it without difficulty. If you want, a time-delay loop can be added between the PRINT statement and RETURN. The ten-line spacing leaves ample room. Perhaps you prefer a little audio reward, which can be used

This program is terse enough to allow for some splendid extra decorations on even the lowest capacity computers. Why not add an exciting reward for a perfect score? This would be an ideal application of one of the extra subroutines printed in the Appendix.

Twenty Questions

```
2 REM  "TWENTY QUESTIONS"
10 PRINT "□":REM  REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
20 PRINT:PRINT:PRINT "TWENTY QUESTIONS"
30 PRINT:PRINT "PRESS RETURN TO START."
40 PRINT:PRINT
50 INPUT "READY";A$
90 DIM DP(20)
100 FOR DR=1 TO 20
110 R=INT(RND(1)*20)+1
120 IF DP(R)=1 THEN 110
130 RESTORE
140 FOR D=1 TO R
150 READ I$,K$
160 NEXT D
170 DP(R)=1
180 GOSUB 500
190 NEXT DR
200 PRINT:PRINT
210 PRINT "THAT'S ALL."
220 PRINT "YOU GOT";20-V;"OUT OF 20"
230 PRINT "CORRECT."
240 IF V<9 THEN PRINT "GOOD WORK!"
250 IF V<15 AND V>8 THEN PRINT "YOU NEED TO STUDY."
260 IF V<21 AND V>14 THEN PRINT "AN AWFUL SCORE!"
270 PRINT:INPUT "TRY AGAIN";A$
280 IF A$="Y" OR A$="YES" THEN RUN
290 PRINT:PRINT "BYE."
300 END
500 FOR WA=1 TO 3
510 RESTORE
520 R=INT(RND(1)*20)+1
530 FOR D=1 TO R
540 READ W$(WA),Z$
550 NEXT D
560 IF W$(WA)=I$ THEN 510
570 NEXT WA
580 IF W$(2)=W$(1) OR W$(2)=W$(3) THEN 500
590 IF W$(1)=W$(3) THEN 500
600 PRINT "□":REM  REPLACE THIS LINE AS ABOVE.
610 PRINT:PRINT:PRINT "QUESTION #";DR
620 PRINT:PRINT "WHO WAS"
630 PRINT K$;"?"
640 PRINT:PRINT
650 R=INT(RND(1)*3)+1
660 FOR P=1 TO 3
670 IF R=P THEN PRINT I$
680 PRINT W$(P)
690 NEXT P
700 PRINT:INPUT "YOUR CHOICE";A$
710 IF A$=I$ THEN PRINT"***CORRECT***":RETURN
```

```
730 PRINT "WAS ";K$;"."
740 FOR T=1 TO 1500:NEXT T
745 V=V+1
750 RETURN
1000 DATA "ABRAHAM","AN ELDERLY FATHER"
1010 DATA "ABSALOM","DAVID'S REBELLIOUS SON"
1020 DATA "AHAB","JEZEBEL'S HUSBAND"
1030 DATA "CALEB","A HEBREW SPY IN CANAAN"
1040 DATA "CYRUS","KING OF PERSIA"
1050 DATA "DANIEL","WITH BABYLONIAN LIONS"
1060 DATA "ELIJAH","IN A FLAMING CHARIOT"
1070 DATA "ESAU","JACOB'S HAIRY BROTHER"
1080 DATA "HEROD","KING AT JESUS' BIRTH"
1090 DATA "ISAAC","ALMOST SACRIFICED"
1100 DATA "JACOB","A BIRTHRIGHT THIEF"
1110 DATA "JONATHAN","DAVID'S BEST FRIEND"
1120 DATA "JOSHUA","MOSES' SUCCESSOR"
1130 DATA "NICODEMUS","JESUS' NIGHT VISITOR"
1140 DATA "PILATE","A ROMAN PROCURATOR"
1150 DATA "QUIRINIUS","CHRISTMAS GOVERNOR"
1160 DATA "SAMSON","A STRONG MAN"
1170 DATA "SAMUEL","A HELPER FOR ELI"
1180 DATA "SOLOMON","A WISE KING"
1190 DATA "TIMOTHY","A YOUNG CHRISTIAN"
```

15

WHO WAS THAT?

Here is a game that catches on instantly and has proven to be popular among all age groups. Adults love it. Children become engrossed, provided they have been taught enough Bible to be familiar with the major characters.

The computer will flash the name of a person from the Bible on the screen for a split second. The player is then asked to wait for a moment before typing the name perceived. Some names are easy, like *Paul*. Others are more difficult, like *Melchizedek*. It will surprise most people when they discover how brief an instant it takes

able to spell it correctly. The computer will not accept an almost right answer.

The player has a complete control over the degree of difficulty presented by each game. At the very beginning the computer asks for input (line 190) from 1 to 10. This difficulty choice is assigned to the variable *D* and remains in effect for all ten rounds. A low number results in longer displays of each name; a high number makes them shorter. This is handled by the formula in line 310. Note that by dividing the loop by Difficulty + Round, each flash gets a little shorter than the one before. You could accentuate this effect by multiplying *R* by a number of your choice.

Watch the effects of line 310 closely when you test the program on your computer. As mentioned in chapter 2, there is a noticeable variation in the speed with which the various brands of computer run through a timing loop. Watch the screen. If you can't see any name at all when you try it at difficulty level 10, then change the 700 in line 310 to 1000, or even 1500. If you think the name appears too long for hot-shot players, then reduce the 700 to about 500. To make the game exciting, it is best to err on the side of displays that are too quick. After a game or two, a player who is really concentrating can recognize in a flash a biblical name that spectators may not see at all. Remember: test and change line 310 until you get it right.

The other factor that will determine the success of the game is the quality of the names you store in your DATA bank. The fifty-three names in the printed version were typed in as they occurred to the author. You can throw them all away, if you like, and enter names of your own selection. Perhaps you would prefer to make an exclusively New Testament version of this game. Another fine specialization would be to list only those people whom Jesus met, or persons mentioned in the letters of Paul. There is just one critical thing to take care of if you change the DATA. Be sure to count *exactly* how many names you have stored, and enter that total as the number inside the last parenthesis in line 230. Where it now says 53, you must put your number. Of course, it can be any number you choose. The more names you have the patience to enter, the more interesting the game. Once in a while the random number generator in line 230 that selects a name from the DATA will stop at the same place twice. The larger the final number, the less likely this will be to happen. It can be prevented altogether by placing a "flag" on used DATA. You can see how this is done in lines 120 and 170 of "Twenty Questions."

which round will end the game. At the present, the magic number is round 10. Feel free to change it to suit your needs. For young children with shorter attention spans, five rounds may be enough. For adults, you may be successful with twenty. The lines for ending the game must be listed in two areas because there is no way of telling whether the last try will be a right or a wrong answer.

If you change the number of rounds, you will also need to change the print statement in line 920 to reflect your alteration.

This game is a good one for demonstrating how easy it is to customize any of these programs. In addition to the suggestions above, there are almost limitless possibilities for change and decoration. For instance, any of the language in the correct answer responses (500–590) or the wrong answer responses (600–690) can be rewritten to your own standards. This would be a great place to introduce a string variable of either the player's or the computer's choosing. Each response could then be a little different after each round. That would make for fresh, good fun.

If your computer has the capacity for it, a variety of reward opportunities could be offered, depending upon the final score in line 960. A top score could trigger an opening into a truly spectacular reward that will make the players strive repeatedly until they achieve it. They will feel no pain at being required to learn to spell these biblical names correctly.

A few minutes spent in the easy work of centering and spacing the printed lines will greatly improve the appearance of the game to the player. Pay special attention to cleaning up the lines of instructions.

When the game is printed in Microsoft BASIC, each name will flash in the upper left-hand corner of your CRT. By using your computer's TAB commands you can guide the names to the center of the screen. If you do that, it will be best to place the "PLEASE WAIT . . ." statement in the same location.

Who Was That?

```
10 REM  "WHO WAS THAT?"
100 PRINT "I WILL FLASH A NAME ON"
110 PRINT "THE SCREEN FOR ONLY AN"
120 PRINT "INSTANT.  YOU MUST"
130 PRINT "RECOGNIZE IT AND SPELL"
140 PRINT "IT CORRECTLY."
150 FOR T=1 TO 2500:NEXT T
160 PRINT "CHOOSE YOUR LEVEL OF":PRINT "DIFFICULTY (1 TO 10)."
170 PRINT "QUICKER FLASHES AND":PRINT "HIGHER SCORES COME"
180 PRINT "WITH HIGHER CHOICE."
190 INPUT "WHICH LEVEL";D
200 IF D=0 OR D>10 THEN 190
210 PRINT"□":REM  REPLACE THIS LINE WITH CLEAR SCREEN COMMAND FOR YOUR COMPUTER
220 R=0:C=0
230 B=INT(RND(1)*53)+1
240 R=R+1
250 RESTORE
260 FOR Z=1 TO B
270 READ B$
280 NEXT Z
300 PRINT B$
310 FOR N=1 TO INT(700/D+R)
320 NEXT N
330 GOSUB 400
340 PRINT "NOW TELL ME WHO WAS"
350 INPUT "THAT";G$
360 IF G$=B$ THEN GOTO 500
370 IF G$<>B$ THEN GOTO 600
400 FOR W=1 TO 150
410 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
420 PRINT "PLEASE WAIT. . ."
430 NEXT W
440 RETURN
500 PRINT "YES!  YOU GOT IT!"
510 PRINT "GOOD WORK!"
520 GOSUB 700
530 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
540 IF R=10 THEN 900
550 PRINT:PRINT "GET READY, HERE":PRINT "COMES ANOTHER ONE!"
560 GOSUB 700
570 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
580 C=C+1
590 GOTO 220
600 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
610 PRINT "SORRY, BIBLE SCHOLAR,"
620 PRINT "YOU SPELLED IT WRONG."
630 GOSUB 700
640 PRINT "□":REM  REPLACE THIS LINE AS ABOVE
650 IF R=10 THEN 900
660 PRINT:PRINT "WATCH OUT! HERE COMES":PRINT "ANOTHER NAME!"
```

```
680 PRINT "Q":REM  REPLACE THIS LINE AS ABOVE
690 GOTO 220
700 FOR T=1 TO 1000: NEXT T
710 RETURN
900 PRINT "Q":REM  REPLACE THIS LINE AS ABOVE
910 PRINT:PRINT:PRINT:PRINT "YOU ANSWERED";C
920 PRINT "OUT OF 10 CORRECTLY!"
930 A=INT(C/R*100)
940 PRINT "YOUR AVERAGE:";A;"%"
950 PRINT "DIFFICULTY LEVEL:";D
960 PRINT "FINAL SCORE:";A*D
970 END
1000 DATA "NOAH","ABRAHAM","ISAAC","JACOB","JOSEPH","MOSES","AARON"
1010 DATA "JOSHUA","SAUL","DAVID","SOLOMON","REHOBOAM","JEROBOAM"
1020 DATA "JEHOSHAPHAT","AHAB","JEHORAM","AHAZIAH","UZZIAH","AHAZ"
1030 DATA "JEREMIAH","NAHUM","MALACHI","ISAIAH","PAUL","PETER"
1040 DATA "JAMES","JOHN","MARY","JOSEPH","PILATE","JUDAS","NICODEMUS"
1050 DATA "TIMOTHY","SALOME","HEROD","THOMAS","NATHANAEL","EUTYCHUS"
1060 DATA "ABSALOM","BARABBAS","BARTHOLOMEW","BENJAMIN","BOAZ"
1070 DATA "HERODIAS","EPAPHRAS","ESAU","MELCHIZEDEK","MARTHA"
1080 DATA "MATTHIAS","GIDEON","SAMSON","ELISHA","MIRIAM"
```

As a demonstration of how this plain program can be enhanced for a particular machine, a second printout is listed on the following pages. This example was prepared for the VIC-20, with a Super Expander cartridge inserted.

The odd-looking characters in line 10 are reverse *Q* symbols, which indicate cursor down. The reverse *R* and reverse _ print the title negative.

Notice how the music code, which begins with a reverse *F* obtained by depressing the CONTROL key and the arrow before the number 1, is split up into two parts in lines 505 and 515. The result is that the language in line 510, "GOOD WORK!" pops on the screen while the music is playing.

Pay special attention to the way line 282 prepares the player for where to expect the name to appear. This is a very nice touch.

Who Was That? (VIC-20 version)

```
1 REM "WHO WAS THAT?" (VIC 20 VERSION)
10 PRINT "♥QQQQQQ    ■WHO WAS THAT?■"
20 PRINT "     [illegible]"
30 PRINT "QQ"
100 PRINT "I WILL FLASH A NAME ON"
110 PRINT "THE SCREEN FOR ONLY AN"
120 PRINT "INSTANT.    YOU MUST"
125 PRINT
130 PRINT "RECOGNIZE IT AND SPELL"
140 PRINT "IT CORRECTLY."
150 FOR T=1 TO 3000:NEXT T
155 PRINT "♥"
160 PRINT "CHOOSE YOUR LEVEL OF":PRINT "DIFFICULTY (1 TO 10)."
165 GOSUB 700
169 PRINT"QQ"
170 PRINT "QUICKER FLASHES AND":PRINT "HIGHER SCORES COME"
180 PRINT"WITH HIGHER CHOICE."
185 GOSUB 700
189 PRINT"QQ"
190 INPUT "WHICH LEVEL";D
192 POKE 36878,15
194 FOR S=130 TO 254
195 POKE 36876,S
196 FOR V=1 TO 20: NEXT V
197 NEXT S
198 POKE 36878,0
199 POKE 36876,0
200 IF D=0 OR D>10 THEN 190
210 PRINT "♥"
220 R=0: C=0
230 B=INT(RND(1)*53)+1
240 R=R+1
250 RESTORE
260 FOR Z=1 TO B
270 READ B$
280 NEXT Z
282 IF R=1 THEN PRINT "QQQQQQQ (NAME APPEARS HERE)"
283 IF R=1 THEN GOSUB 700
285 PRINT "♥"
290 PRINT "QQQQQQQQ"
300 PRINT "        ";B$
310 FOR N=1 TO INT(700/D+R)
320 NEXT N
330 GOSUB 400
340 PRINT "QQQQQQNOW TELL ME WHO WAS"
350 INPUT "THAT";G$
352 POKE 36878,15
353 FOR S=1 TO 25
354 POKE 36875,INT(RND(1)*128)+128
355 FOR M=1 TO 10:NEXT M
```

```
357 POKE 36875,0
358 POKE 36878,0
360 IF G$=B$ THEN GOTO 500
370 IF G$<>B$ THEN GOTO 600
400 FOR W=1 TO 150
410 PRINT "■"
415 PRINT "■■■■■■■■■■"
420 PRINT "   PLEASE WAIT. . . "
430 NEXT W
435 PRINT "■"
440 RETURN
500 PRINT "YES!  YOU GOT IT!"
505 PRINT "■V9 93 02 T2 0 03 CEQR
510 PRINT "GOOD WORK!"
515 PRINT "■E T5 0
520 GOSUB 700
530 PRINT "■"
540 IF R=10 THEN 900
550 PRINT "■■■■■■■  GET READY, HERE":PRINT "COMES ANOTHER ONE"
560 GOSUB 700
570 PRINT "■"
580 C=C+1
590 GOTO 230
600 PRINT "■"
610 PRINT "SORRY, BIBLE SCHOLAR,"
620 PRINT "YOU SPELLED IT WRONG.":PRINT "■■■IT WAS ";B$
621 POKE 36877,220
622 FOR E=15 TO 0 STEP-2
623 POKE 36878,E
624 FOR S=1 TO 225: NEXT S
625 NEXT E
626 POKE 36877,0
627 POKE 36878,0
630 GOSUB 700
640 PRINT "■"
650 IF R=10 THEN 900
660 PRINT "■■■■■■■WATCH OUT! HERE COMES":PRINT" ANOTHER NAME!"
670 GOSUB 700
680 PRINT "■"
690 GOTO 230
700 FOR T=1 TO 1000: NEXT T
710 RETURN
900 PRINT "■"
910 PRINT "■■■YOU ANSWERED";C
920 PRINT "OUT OF 10 CORRECTLY!"
930 A=INT(C/R*100)
940 PRINT "YOUR AVERAGE:";A;"%"
950 PRINT "DIFFICULTY LEVEL:";D
960 PRINT "FINAL SCORE:";A*D
970 END
1000 DATA "NOAH","ABRAHAM","ISAAC","JACOB","JOSEPH","MOSES","AARON"
1010 DATA "JOSHUA","SAUL","DAVID","SOLOMON","REHOBOAM","JEROBOAM"
1020 DATA "JEHOSHAPHAT","AHAB","JEHORAM","AHAZIAH","UZZIAH","AHAZ"
1030 DATA "JEREMIAH","NAHUM","MALACHI","ISAIAH","PAUL","PETER"
1040 DATA "JAMES","JOHN","MARY","JOSEPH","PILATE","JUDAS","NICODEMUS"
1050 DATA "TIMOTHY","SALOME","HEROD","THOMAS","NATHANAEL","EUTYCHUS"
1060 DATA "ABSALOM","BARABBAS","BARTHOLOMEW","BENJAMIN","BOAZ"
1070 DATA "HERODIAS","EPAPHRAS","ESAU","MELCHIZEDEK","MARTHA"
1080 DATA "MATTHIAS","GIDEON","SAMSON","ELISHA","MIRIAM"
```

16

FAMOUS BIBLE WOMEN

Here is a beautiful program! It is the result of much refinement and simplification. It performs some very complicated tricks with a minimum of memory consumption; and if you want, you can "crunch" a few of the lines together for additional savings. It is presented in its present form for ease of reading and accurate copying.

Begin with an examination of the DATA. Each line contains three elements:

A$(1) All the letters of a name, scrambled.

A$(2) Her name.

A$(3) A brief clue regarding the woman.

These trios of information are listed for thirty-one different biblical women. You can add to or subtract from the list all you want. Be sure to multiply your total number of women by three and put that number in parentheses in line 90.

By changing the DATA, you can change the entire game. You could adapt this program for just about any biblical subject, from "Old Testament Heroes" to "Young People of the Bible." Notice that the trick to scrambling a name is to make the letters read like some other word. This greatly increases the player's difficulty and fun.

In experimenting with this program, be sure to keep line 90 ahead of any return point for the next name. The game can be DIMensioned only once for each player. Alarm bells will ring in your computer if you pass this line a second time.

The fun in playing this game comes from getting higher scores by not asking for help. Every time the player asks for a clue, it costs five points. There is just one clue per name and there is no guarantee that it will always help. Of course, if the name must be skipped entirely by typing "Can't", or any other nonsense, and pressing the return key, the loss is ten points.

As with any game using DATA, spelling is of primary importance. The computer simply will not accept an almost right answer. It must be perfect. Even a space before the first letter of the name will result in an error. If you have made any typographical errors when you entered the DATA, then no one will ever be able to get that name correct. Proofread your DATA. Let a second pair of eyes check the DATA over before you turn the game loose for play. You can mess up the scrambled name any way you like with no harm done. Just be sure to get all of the necessary letters on the screen.

Lines 340 and 350 keep the player informed. It is always a good idea to let the player know how long the game is likely to last and how the score stands.

Line 1010 presents an idea you may want to add to the other games. It gives the player an option of skipping the instructions when taking another turn. Lines 1030 and 1040 will kick the run back to a point either before or after the instructions, depending upon the player's choice. Incidentally, your computer may require a GOTO after the THEN in these two and other similar lines. See the introductory chapters.

Line 720 waits patiently for ten names to be selected from DATA and acted upon by the player. When R = 10 it causes the computer to bypass line 730 and enter the end of game routine. Line 810 prevents a negative score, which is possible if a player asks for clues and still fails to unscramble any names.

The section beginning with line 900 gives the player an option to attempt ten more names or to quit. If the player quits, the computer will rapidly display the word "CHICKEN" many times until the screen is full. The semicolon after the quotation mark in line 960 prevents the words from appearing in a narrow column on the left side of the screen. Once this fireworks display starts, it will run forever. The only way to stop it is to press a "STOP" key or to turn off the computer.

"Famous Bible Women" has found instant appeal with all players. Adults will spend a very long time with it if left undisturbed.

Famous Bible Women

```
10 REM  "FAMOUS BIBLE WOMEN"
90 DIM Z(93)
100 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
110 PRINT:PRINT:PRINT "I WILL SCRAMBLE THE"
120 PRINT"NAMES OF 10 BIBLE WOMEN."
130 PRINT:PRINT "ENTER YOUR GUESS, OR"
140 PRINT "TYPE 'CLUE' FOR HELP."
150 PRINT:PRINT "CLUES COST 5 POINTS."
160 PRINT:PRINT "RIGHT ANSWER WINS 10."
170 PRINT:PRINT "HIGHEST SCORE=100."
180 PRINT:PRINT "READY? PRESS RETURN."
190 INPUT P
210 S=0:R=0
220 PRINT "□":REM  REPLACE THIS LINE AS BEFORE.
230 RESTORE
240 N=INT(RND(1)*31)
250 W=N*3
260 IF W=0 THEN 310
270 IF Z(W)=1 THEN 240
280 Z(W)=1
290 FOR WC=1 TO W
300 READ Z$: NEXT WC
310 FOR X=1 TO 3
320 READ A$(X): NEXT X
330 R=R+1
340 PRINT:PRINT "WOMAN NUMBER:";R
350 PRINT:PRINT "CURRENT SCORE:";S
360 PRINT:PRINT:PRINT A$(1)
370 PRINT:PRINT "TRY IT OR ASK FOR CLUE"
380 INPUT "YOUR CHOICE";X$
400 IF X$="CLUE" THEN 500
410 IF X$<>A$(2) THEN PRINT:PRINT "WRONG! IT WAS ";A$(2):GOTO 600
420 IF X$=A$(2) THEN PRINT "RIGHT!":GOTO 700
500 PRINT:PRINT "CLUE:";A$(3)
510 S=S-5
520 GOTO 380
600 FOR T=1 TO 1700: NEXT T
610 IF R=10 THEN 800
620 GOTO 220
700 FOR T=1 TO 700: NEXT T
710 S=S+10
720 IF R=10 THEN 800
730 GOTO 220
800 PRINT "□":REM  REPLACE THIS LINE AS ABOVE.
810 IF S<0 THEN S=0
820 PRINT:PRINT:PRINT "FINAL SCORE:";S
830 FOR T=1 TO 1000: NEXT T
840 PRINT:PRINT:PRINT
900 PRINT "DO YO WANT TEN MORE?"
```

```
910 INPUT "'Y' OR 'N'";P$
920 IF P$="Y" THEN 1000
930 IF P$="N" THEN 950
940 GOTO 910
950 PRINT "□":REM  REPLACE THIS LINE AS ABOVE.
960 PRINT "CHICKEN!";
970 GOTO 960
1000 PRINT "□":REM  REPLACE THIS LINE AS ABOVE."
1010 PRINT:PRINT:PRINT "WANT INSTRUCTIONS?"
1020 INPUT "'Y' OR 'N'";I$
1030 IF I$="Y" THEN 100
1040 IF I$="N" THEN 210
1050 GOTO 1020
2000 DATA "NAAN","ANNA","N.T. PROPHETESS"
2010 DATA "THEBABASH","BATHSHEBA","DAVID'S DESIRE"
2020 DATA "HOLEC","CHLOE","CORINTHIAN CHRISTIAN"
2030 DATA "DIALAUC","CLAUDIA","ROMAN CHRISTIAN"
2040 DATA "REBAHDO","DEBORAH","REBECCA'S NURSE"
2050 DATA "HALLIDE","DELILAH","BETRAYED LONG HAIR"
2060 DATA "NADIH","DINAH","JACOB'S DAUGHTER"
2070 DATA "CARSOD","DORCAS","NEEDLE AND THREAD"
2080 DATA "RAIDLUS","DRUSILA","GOVERNOR'S WIFE"
2090 DATA "STABEHELI","ELISABETH","JOHN'S MOTHER"
2110 DATA "CUENIE","EUNICE","TIMOTHY'S MOTHER"
2120 DATA "ROMEB","GOMER","HOSEA'S WIFE"
2130 DATA "DASHROIE","HERODIAS","JOHN'S ENEMY"
2140 DATA "BEZELEJ","JEZEBEL","WICKED QUEEN"
2150 DATA "SOIL","LOIS","TIMOTHY'S GRANDMOTHER"
2160 DATA "DAILY","LYDIA","FIRST EUROPEAN CONVERT"
2170 DATA "THAARM","MARTHA","MARY'S SISTER"
2180 DATA "NEARAMYMADLEG","MARY MAGDALENE","7 DEMONS"
2190 DATA "RMMIIA","MIRIAM","MOSES' SISTER"
2200 DATA "MANOI","NAOMI","RUTH'S MOTHER-IN-LAW"
2210 DATA "LARCEH","RACHEL","JACOB'S WIFE"
2220 DATA "HARBA","RAHAB","HARLOT"
2230 DATA "CABEREC","REBECCA","ISAAC'S WIFE"
2240 DATA "THRU","RUTH","FAMOUS IN-LAW"
2250 DATA "MALSOE","SALOME","DANCED"
2260 DATA "PRASAPIH","SAPPHIRA","WIFE OF ANANIAS"
2270 DATA "RAASH","SARAH","ABRAHAM'S WIFE"
2280 DATA "NUNSASA","SUSANNA","HELPED JESUS (LK 8:3)"
2290 DATA "MARTA","TAMAR","DAVID'S DAUGHTER"
2300 DATA "TISAHV","VASHTI","QUEEN OF PERSIA"
2310 DATA "LLIZAH","ZILLAH","WIFE OF LAMECH"
```

17

FORTY-TWO KINGS

Here are three games in one, plus an introduction. There is no reason at all to be defeated by a long program like this. It took days to design, and it is permissible for you to take days to enter it. The beauty of it is that you can have a playable game right from the start, and then add two, three, or four segments as time goes by. Before we get into how to do that, let's see what kind of fun you can have with this one.

Of all the memory work in the Bible that defeats nearly every-

Some of these kings reigned an extraordinarily long time (Manasseh, fifty-five years), and others barely felt the weight of the crown (Jehoahaz, three months). The Bible has nothing nice to say about any of the kings of the north, and the kings of the south get mixed reviews. Casual Bible readers have been known to become confused wandering through such a wilderness of odd names.

This game is intended to desensitize us. By the time a player can win any portion of it, the names have become as familiar as those of players on a ball team. It would be an exaggeration to say that the game takes *all* the pain out of learning their names and the order of their succession, but it certainly goes a long way in that direction. The computer never gets angry or loses patience. It is forever optimistic and helpful.

It is highly unlikely that any player will win any part beyond the introduction of this game on the first few attempts. A woman in her middle years with whom we tested the game was able to reach nine kings with about fifteen minutes of playing time. The next day, she returned to the challenge and successfully made it to thirteen. The same lure that draws the arcade patron and his quarters back for another round of Donkey Kong™ is at work in "Forty-two Kings." It can be done, but not without practice.

The introduction of the game insists that the player get the big three kings in order and properly spelled. Once Saul, David, and Solomon are firmly established, the computer gives the player a choice of taking on either the nineteen kings of Israel or the twenty kings of Judah. It then presents them all, clearly, in order, slowly. The screen is erased and the first name only is flashed for the player to absorb. The computer then asks the player to type that single name. If the player can't remember it, or misspells it, the computer patiently displays a reminder. By typing "Help", the player can have a reminder at any point, with no penalty. The list will always be presented only up to the present point of challenge.

With success on the first name, the computer will then automatically display the first two names and ask that they be entered in order. Then it lists three, four, five, and so on until the player can master the challenge and list every name in order, properly spelled.

Learning comes with surprising speed. By the time the player is working in the region of fifteen and sixteen names, the first dozen are known very well. Excitement mounts near the end when the discovery is made that Pekahiah is followed by Pekah, and Jehoiakim by Jehoiachin.

display is set off as a reward, and the player is challenged to tackle the other. The computer actually remembers which kingdom the player has selected with SR$ in lines 620, 3680, and 3690. If the player answers "No", the only response is a quiet "Bye." If the answer is "Yes", then the program leaps back to the beginning of the other list and starts an identical process of enjoyable education.

If, after a number of days, a player successfully completes both kingdoms, there is an invitation to accept the super challenge: all forty-two kings, in order. This is the third level on which the game can be played, and it will be won by only a few. Because of the size of the challenge, provision has been made for the attachment of a knockout reward routine. You can invent one of your own, or use one listed in the Appendix. It ought to be spectacular.

Perhaps you are put off by the task of entering a program that is as long as this. Don't be! The beauty of this one is that you don't have to put it all in to get something back out. You can start playing it almost at once. If you like it, you will be delighted to add more later. If it is not your cup of tea, you have wasted very little of your time. If you store your work on cassette or disk you will be able to build up a game of truly gargantuan proportions that requires surprisingly little computer memory.

Here is how to get started. Follow these directions very carefully. First, enter the DATA. You will find it near the end of the printed listing, starting with line 5000. The REMarks are there to help you find your way around. You can omit them all in your version. Aside from the usual care and accuracy, notice the quotation marks around "JEROBOAM II". They are necessary to keep the space between the name and the numeral intact. Other than this one instance, there are no spaces anywhere else in the DATA.

Once you have the DATA in place and have double-checked your spelling and commas, return to the beginning of the program. Enter the subroutines in the low numbers. Even the screen-clearing command has been made a subroutine in this game. It is used so often, we put it here for tidiness. If you can type something shorter than GOSUB 10 and get a clear screen, don't hesitate to eliminate lines 10 and 11 and replace each GOSUB 10 you come to with your preferred code. The other two subroutines are simple time-delay loops, a short one and a longer one. These are called on many times, and the numbers 6 and 8 are quick and easy to enter.

H$, T$, and J$ in lines 70–90 keep you from having to repeat a lot of PRINT statements later. By the time the program has run to

down through line 650 and you will have an easy, playable game. Go ahead and RUN it when you have this much done.

Once you see what you are in for, you will not want to stop until you have added at least one of the three major blocks of game. Choose either Israel or Judah. It will make no difference at all. If you want to try Israel, add lines 660 through 980. If it is Judah, then add lines 2000 through 2320. You will need to add a temporary END statement as an extra line for either if you want to RUN it again in this expanded state.

The next part to add would be the alternate kingdom. Again, you can RUN it for test purposes at this stage. Then add the win routine from lines 3500 through 3730. Finally, tack on the big game of all forty-two names that stretches from line 3800 to the DATA.

As you see, you can take a week or more to put this one in storage, and you can play at it all along the way. Once you have an idea of the scope of this one, you can take the final satisfaction of adding the huge reward suggested in the REM statement in line 4070.

There are some very delicate features in this game that save much memory and paper. Notice that the working game itself is a series of nested loops. If you study lines 890 through 940 you will discover everything necessary for putting the player to the test. These same lines are called upon over and over until each name in the list has been completed.

Forty-two Kings

```
1 REM  "42 KINGS"
5 GOTO 70
6 FOR T=1 TO 2000:NEXT T
7 RETURN
8 FOR T=1 TO 1000:NEXT T
9 RETURN
10 PRINT"⌧":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
11 RETURN
70 H$="HIGHEST NUMBER>"
80 T$="TYPE NAMES AND PRESS  RETURN AFTER EACH."
90 J$="(IF YOU GET IN A JAM, TYPE 'HELP'.)"
100 DIM K$(42)
110 FOR R=1 TO 42
120 READ K$(R)
130 NEXT R
140 GOSUB 10
150 PRINT:PRINT "AH!  YOU ARE HERE!"
160 GOSUB 8
170 PRINT:PRINT "TELL ME YOUR NAME.":PRINT
180 INPUT N$
190 PRINT:PRINT "YOU WILL FIND ME TO BE A PATIENT INSTRUCTOR, ";N$;"."
200 GOSUB 6
210 PRINT:PRINT "I WILL TELL YOU THINGS AND ASK QUESTIONS.":PRINT:PRINT
220 GOSUB 6
230 INPUT "SHALL WE START";I$
240 GOSUB 10
250 PRINT:PRINT "SAUL WAS KING FIRST IN ISRAEL. THE 12 TRIBES WERE UNITED."
260 GOSUB 6
270 PRINT:PRINT "SAUL WENT MAD."
275 GOSUB 8
280 K$="THE FIRST THREE KINGS "
290 PRINT:PRINT "DAVID BECAME KING."
300 GOSUB 8
310 PRINT:PRINT "DAVID'S SON, "K$(3);", REIGNED AFTER HIM."
320 GOSUB 6
330 PRINT:PRINT "THE FIRST THREE KINGS WERE:"
340 GOSUB 6
350 PRINT:PRINT
360 FOR L=1 TO 3
370 PRINT TAB(10);K$(L)
380 GOSUB 8
390 NEXT L
400 GOSUB 6
410 GOSUB 10
420 PRINT:PRINT:PRINT "ENTER THE NAMES OF THE FIRST 3 KINGS,"
425 PRINT "ONE AT A TIME, IN ORDER."
430 FOR L=1 TO 3
440 INPUT I$
450 IF I$<>K$(L) THEN 480
460 NEXT L
```

```
480 PRINT:PRINT "I'M SORRY, ";N$;"."
490 GOSUB 8
500 PRINT:PRINT "WE CAN'T GO ON UNTIL YOU GET THEM STRAIGHT."
510 GOSUB 6
520 GOTO 240
530 PRINT:PRINT "YES! YOU LISTEN WELL, ";N$;"!"
540 GOSUB 8
550 PRINT:PRINT:PRINT "AFTER SOLOMON A CIVIL WAR DIVIDED THE KINGDOM."
560 GOSUB 6
570 PRINT:PRINT "THE NORTH WAS ISRAEL."
580 GOSUB 8
590 PRINT:PRINT "THE SOUTH WAS JUDAH.":PRINT:PRINT
600 GOSUB 8
610 PRINT:PRINT "WHICH KINGS SHALL WE STUDY NOW, ";N$;"?":PRINT
620 INPUT "ISRAEL OR JUDAH";SR$
630 IF SR$="ISRAEL" THEN 660
640 IF SR$="JUDAH" THEN 2000
650 GOTO 620
659 REM ------------------------------< ISRAEL >-------------------------
660 GOSUB 10
670 GOSUB 8
680 PRINT:PRINT "THE 19 KINGS OF ISRAEL"
690 GOSUB 8
700 FOR L=4 TO 22
710 PRINT:PRINT TAB(4);L-3;:IF L-3<10 THEN PRINT " ";
711 PRINT "  ";K$(L)
720 GOSUB 8
730 NEXT L
740 GOSUB 8
750 PRINT:PRINT "DON'T BE AFRAID, ";N$;"!"
760 GOSUB 8
770 PRINT:PRINT "YOU CAN LEARN THEM IF I HELP YOU."
780 GOSUB 6
790 GOSUB 10
800 PRINT:PRINT:PRINT "LET'S TAKE THEM ONE AT A TIME. . ."
810 GOSUB 8
820 PRINT:PRINT
830 FOR M=4 TO 22
840 FOR L=4 TO M
850 PRINT TAB(4) L-3;:IF L-3<10 THEN PRINT " ";
860 PRINT K$(L)
870 NEXT L
880 GOSUB 6:GOSUB 10:IF S<L-3 THEN S=L-5
890 PRINT:PRINT T$:PRINT:PRINT J$:PRINT
895 PRINT H$;S:PRINT
900 FOR L=4 TO M
910 PRINT "KING #";L-3;
920 INPUT I$
930 IF I$<>K$(L) THEN 840
940 NEXT L
950 GOSUB 10
960 PRINT:PRINT "LET'S ADD ANOTHER!"
970 NEXT M
980 GOTO 3500
1999 REM ------------------------------< JUDAH >-------------------------
2000 GOSUB 10
2010 GOSUB 8
2020 PRINT:PRINT "THE 20 KINGS OF JUDAH"
2030 GOSUB 8
2040 FOR L=23 TO 42
2050 PRINT:PRINT TAB(4);L-22;:IF L-22<10 THEN PRINT " ";
2051 PRINT "  ";K$(L)
2060 GOSUB 8
2070 NEXT L
2080 GOSUB 8
2090 PRINT:PRINT "YOU CAN DO IT, ";N$;"!"
2100 GOSUB 8
2110 PRINT:PRINT "TRUST ME. . . "
```

```
2140 PRINT:PRINT:PRINT "JUST REPEAT WHAT I SAY."
2150 GOSUB 8
2160 PRINT:PRINT
2170 FOR M=23 TO 42
2180 FOR L=23 TO M
2190 PRINT TAB(4) L-22;:IF L-22<10 THEN PRINT " ";
2200 PRINT K$(L)
2210 NEXT L
2220 GOSUB 6:GOSUB 10:IF S<L-22 THEN S=L-24
2230 PRINT:PRINT T$:PRINT:PRINT J$:PRINT
2235 PRINT H$;S:PRINT
2240 FOR L=23 TO M
2250 PRINT "KING #";L-22;
2260 INPUT I$
2270 IF I$<>K$(L) THEN 2160
2280 NEXT L
2290 GOSUB 10
2300 PRINT:PRINT "TRY ONE MORE..."
2310 NEXT M
2320 GOTO 3500
3499 REM ------------------------------< WIN >--------------------------
3500 GOSUB10
3510 FOR H=1 TO 350
3520 PRINT "HOORAY! ";N$;" DID IT!";
3530 NEXT H
3540 GOSUB 10
3550 PRINT:PRINT:PRINT "I KNEW YOU COULD DO IT!";
3560 FOR H=1 TO 1000
3570 PRINT "!";
3580 NEXT H
3600 PRINT:PRINT ". . . WITH MY HELP, OF COURSE."
3610 GOSUB 6
3620 GOSUB 10
3630 PRINT:PRINT:PRINT "DO YOU KNOW THE OTHER KINGS?"
3640 INPUT "Y OR N";I$
3650 IF I$="Y" THEN 3700
3660 IF I$="N" THEN PRINT "NOW IS A GOOD TIME TO LEARN THEM!":GOSUB 8:GOTO 3680
3670 GOTO 3640
3680 IF SR$="JUDAH" THEN S=0:GOTO 660
3690 IF SR$="ISRAEL" THEN S=0:GOTO 2000
3700 PRINT:PRINT "WOULD YOU DARE ATTEMPT THEM ALL?"
3710 INPUT "Y OR N";I$
3720 IF I$="Y" THEN S=0:GOTO 3800
3730 IF I$="N" THEN PRINT "BYE.":END
3799 REM ------------------------------< ALL >--------------------------
3800 GOSUB 10:GOSUB 8
3820 PRINT:PRINT "**FORTY-TWO KINGS**":PRINT:PRINT
3830 FOR L=1 TO 42
3840 PRINT:PRINT TAB(4);L;:IF L<10 THEN PRINT " ";
3850 PRINT " ";K$(L):GOSUB 8
3860 NEXT L
3870 GOSUB 8:GOSUB 10
3890 PRINT:PRINT:PRINT "NOW SEE IF YOU CAN PUT THEM ALL IN ORDER."
3890 PRINT "UNITED, ISRAEL, JUDAH!"
3910 FOR M=1 TO 42
3920 FOR L=1 TO M
3930 PRINT TAB(4)L;:IF L<10 THEN PRINT " ";
3940 PRINT K$(L)
3950 NEXT L
3960 GOSUB 8:GOSUB 10:S=L-2
3970 PRINT:PRINT T$:PRINT:PRINT J$
3975 PRINT:PRINT H$;S:PRINT
3980 FOR L=1 TO M
3990 PRINT "KING #";L;
4000 INPUT I$
4010 IF I$<>K$(L) THEN 3920
4020 NEXT L
```

```
4060 PRINT "INCREDIBLE!"
4070 END:REM  HERE ADD HUGE REWARD ROUTINE.
4998 REM ------------------------------< DATA >--------------------------
4999 REM KINGS OF UNITED ISRAEL
5000 DATA SAUL,DAVID,SOLOMON
5009 REM KINGS OF ISRAEL
5010 DATA JEROBOAM,NADAB,BAASHA,ELAH,ZIMRI,OMRI,AHAB,AHAZIAH,JORAM
5020 DATA JEHU,JEHOAHAZ,JOASH,"JEROBOAM II",ZECHARIAH,SHALLUM
5030 DATA MANAHEM,PEKAHIAH,PEKAH,HOSHEA
5039 REM KINGS OF JUDAH
5040 DATA REHOBOAM,ABIJAH,ASA,JEHOSHAPHAT,JEHORAM,AHAZIAH,ATHALIAH
5050 DATA ATHALIA,JOASH,AMAZIAH,UZZIAH,JOTHAM,AHAZ,HEZEKIAH
5060 DATA MANASSEH,AMON,JOSIAH,JEHOIAKIM,JEHOIACHIN,ZEDEKIAH
```

18

CAPTIVE!

Captivity is a theme that runs through the Bible from one end to the other. Consult a concordance of the Bible and you will discover a long list of verses from both the Old Testament and the New Testament that speak of both physical and spiritual captivity.

This game is designed to raise the player's consciousness of the horrors of captivity. It does so even while the player is doing the capturing. In spite of its somber theme, the game is an absorbing one to play. It pits the skill of the player against the programmed intelligence of the computer. In fact, the computer had to be intentionally

make a bad move. This fallibility is derived from lines 400 through 500 and lines 610 through 660. Line 450 brings a phony player into the game and line 650 introduces a phony move. The two random number generators found in lines 410 and 610 complete the scatter-braining of the computer. Without this possibility of error, the computer almost can't be beat. Because of it, the player feels a certain excitement of the chase.

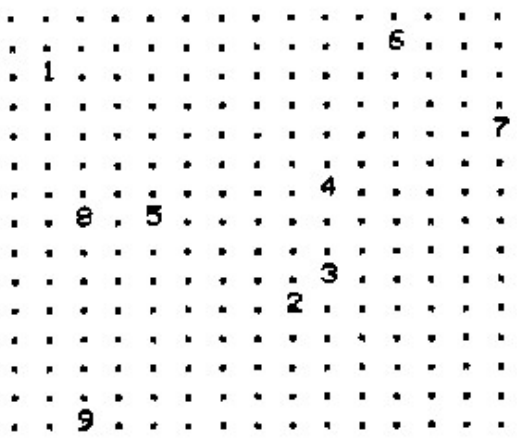
Space does not permit publication of all of the preliminary studies that helped perfect "Captive!" One example is included for its educational value to the users of this book. This program, and a few of its test runs, will show you how an idea for a game can be thought out and perfected. You will see that the computer was given some choices and asked to indicate what it would do.

Game Development and Test Runs

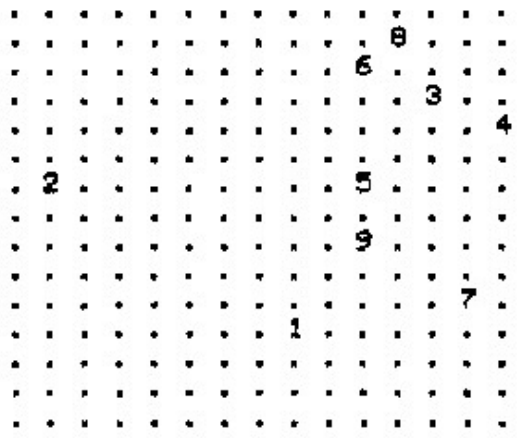
```
10 DIM A(15,15)
20 FOR L=1 TO 9
30 X=INT(RND(1)*15)+1
40 Y=INT(RND(1)*15)+1
50 IF A(X,Y)=1 THEN 60
60 A(X,Y)=L
70 NEXT L
100 FOR Y=1 TO 15
110 FOR X=1 TO 15
120 A$=STR$(A(X,Y))
130 IF A(X,Y)>0 THEN PRINT A$;:GOTO 150
140 PRINT" .";
150 NEXT X
160 PRINT
170 NEXT Y
200 FOR Y=1 TO 15
210 FOR X=1 TO 15
220 IF A(X,Y)>0 THEN P(A(X,Y))=ABS(8-X)+ABS(8-Y)
230 NEXT X:NEXT Y
300 OP=0
310 FOR L=1 TO 9
320 IF P(L)> OP THEN FP=L:OP=P(L)
330 NEXT L
400 FOR Y=1 TO 15
410 FOR X=1 TO 15
420 IF A(X,Y)=FP THEN 440
430 NEXT X:NEXT Y:END
440 X2=SGN(X-8):Y2=SGN(Y-8)
450 PRINT:PRINT "I WOULD MOVE";FP;"FROM"
460 PRINT X;",";Y;"TO";X+X2;",";Y+Y2
```

```
 . 3 . . . . . 8 . .
 . . . . . .
 . . . . . .
 . . . . . 9 . . . . . .
 . . . . . .
 . . . . . .
 . . . . . .
 . . . . . 6 .
 . . . . . 2 . . . . .
 . . . . . 5 .
 . . . . . .
 . . . . . .
 . . . . . 1 7
 . . . . . .
 . . . . 4 . . . . . .

I WOULD MOVE 3 FROM
 2 , 1 TO 1 , 0
```



```
I WOULD MOVE 9 FROM
 3 , 15 TO 2 , 16
```



```
I WOULD MOVE 4 FROM
 15 , 5 TO 16 , 4
```

"Captive!" began as a very slow program in which the computer took up to half a minute to select its move. In its final form, early moves are made in a split second. When you run the program, given below, you will notice that line 320, "I am computing my move," will gradually stay on the screen longer and longer as the game nears a player win. But even as it checks a dwindling supply of options, it seldom needs more than four or five seconds. Except for the random goofs already mentioned, the computer plays with quick, cold, calculating logic.

The lines before 100 give you an idea of how to introduce the game to a player. Lines 15, 20, and 25 present the title in an evocative chain of *O*'s. Put enough spaces after each quotation mark to center the lines on your screen. Be sure to put an identical number

The several lines of instruction should be clear enough for most players. The object is to prevent each "#" from moving in any of the eight directions available to it. The diagonal moves are the most difficult to stop. If a "#" escapes the grid's edge, you lose. The player selects the numbers for two coordinates and a "+" mark indicates the location on the grid. Lines 270, 280, and 290 inform the player of an illegal move and ask for another pair of coordinates. Two objects cannot occupy the same space, and locations off the grid are considered nonexistent. Also, the coordinates must be separated by a comma when they are entered.

Lines 1010 and 1020 provide the player with important locations for the numbered columns across. The easiest way to enter them into the program is to do line 1020 first. Skip four spaces inside the quotation mark, type one to zero, and then add one to five. Next, number line 1010 and space until you get exactly under the zero of line 1020 to start typing six ones.

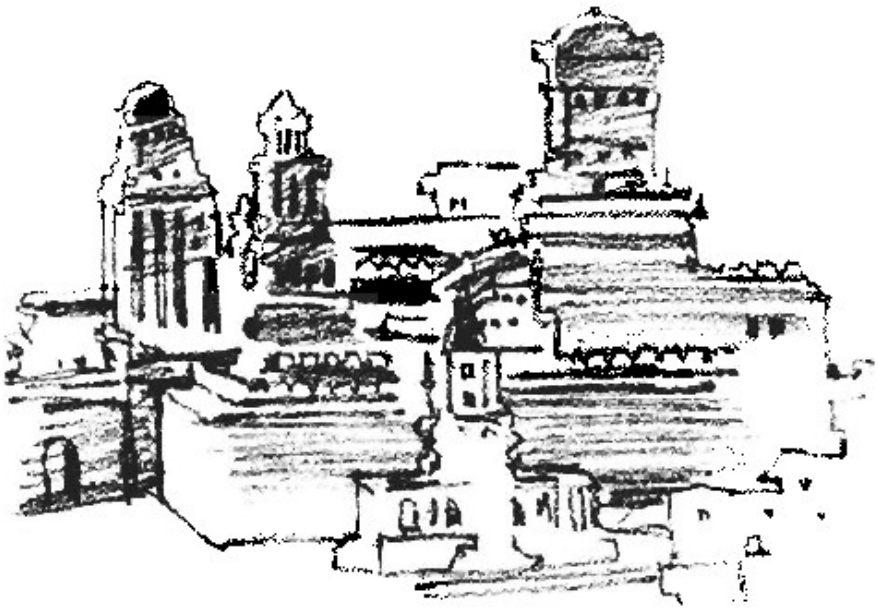
Each game will take about ten to twenty minutes of your time. You will find that it holds your interest all the way to the end and that you will want to try another right away. As you develop strategies, you may begin winning with regularity, but at first you should be prepared to lose a few. Even though it has been tamed a bit, the computer still plays a hot game.

"Captive!" eats up memory. You may be able to LIST it with a computer of 5K or less, but you will need up to 8K for it to RUN. The DIMensioning in lines 100 through 130 guzzles the memory.

Captive!

```
1 REM  "CAPTIVE!"
10 PRINT "□":REM  REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
15 PRINT:PRINT "  ○○○○○○○○○○○○○○○○"
20 PRINT "  ○○○○ CAPTIVE! ○○○○"
25 PRINT "  ○○○○○○○○○○○○○○○○"
30 PRINT:PRINT:PRINT "# MARKS TRY TO ESCAPE"
40 PRINT:PRINT "ENTER COORDINATES TO"
50 PRINT "STOP THEM WITH + MARKS"
60 PRINT:PRINT "   X = ACROSS"
70 PRINT "   Y = DOWN"
80 PRINT:PRINT "IF ANY # ESCAPES GRID, YOU LOOSE..."
90 PRINT:PRINT "WHEN NO MOVE REMAINS, YOU WIN."
95 PRINT:PRINT "PRESS RETURN TO START."
98 INPUT A
100 DIM A(16,16)
110 DIM X(9):DIM Y(9)
120 DIM FD(8)
130 DIM FW(9)
140 PRINT "□":REM  REPLACE AS ABOVE
150 FOR A=7 TO 9
160 FOR B=7 TO 9
170 L=L+1:A(B,A)=1
180 X(L)=A:Y(L)=B
190 NEXT B:NEXT A
200 GOSUB 1000
210 FOR L=1 TO 9
220 IF X(L)>15 OR X(L)<1 THEN 850
230 IF Y(L)>15 OR Y(L)<1 THEN 850
240 NEXT L:C=C+1
250 PRINT:PRINT "PLACE SOLDIER AT WHAT"
260 INPUT "LOCATION (X,Y)";A,B
270 IF A>15 OR A<1 OR B>15 OR B<1 THEN PRINT ">>WHAT?":GOTO 260
280 IF A(A,B)=1 THEN PRINT "SURROUND: DON'T STOMP!":GOTO 260
290 IF A(A,B)=2 THEN PRINT ">>THERE'S ALREADY A SOLDIER THERE":GOTO 260
300 A(A,B)=2
310 GOSUB 1000
320 PRINT:PRINT ">>  I AM COMPUTING MY MOVE."
330 B=0
340 FOR L=1 TO 9
350 A=ABS(X(L)-8)+ABS(Y(L)-8)
360 IF A>B AND FW(L)=0 THEN BW=L:B=A
370 NEXT L
380 FOR L=1 TO 9:FW(L)=0:NEXT L
390 FOR M=1 TO 9
400 A=0:B=0
410 A=INT(RND(1)*16)+1
420 IF A>9 AND FW(BW)=0 THEN W=BW:GOTO 500
430 IF A>9 THEN 410
440 IF FW(A)<>0 THEN 410
```

```
450 W=A
500 A=SGN(X(W)-8):B=SGN(Y(W)-8)
510 IF A= 1 AND B= 0 THEN BD=1
520 IF A= 0 AND B= 1 THEN BD=3
530 IF A=-1 AND B= 0 THEN BD=5
540 IF A= 0 AND B=-1 THEN BD=7
550 IF A= 1 AND B=-1 THEN BD=2
560 IF A=-1 AND B= 1 THEN BD=4
570 IF A=-1 AND B=-1 THEN BD=6
580 IF A= 1 AND B=-1 THEN BD=8
590 FOR L=1 TO 8:FD(L)=0:NEXT L
600 FOR V=1 TO 8
610 A=INT(RND(1)*16)+1
620 IF A>8 AND FD(BD)=0 THEN D=BD:GOTO 660
630 IF A>8 THEN 610
640 IF FD(A)<>0 THEN 610
650 D=A
660 A=0:B=0
670 IF D=1 THEN A= 1
680 IF D=3 THEN B= 1
690 IF D=5 THEN A=-1
700 IF D=7 THEN B=-1
710 IF D=2 THEN A= 1:B= 1
720 IF D=4 THEN A=-1:B= 1
730 IF D=6 THEN A=-1:B=-1
740 IF D=8 THEN A= 1:B=-1
750 IF A(X(W)+A,Y(W)+B)=0 THEN 900
760 FD(D)=1
770 NEXT V
780 FW(W)=1
790 NEXT M
800 PRINT "□":REM  REPLACE AS ABOVE
810 PRINT:PRINT:PRINT "BOY, I BLEW IT!"
820 PRINT:PRINT "YOU WIN."
830 PRINT "IN";C;"ROUNDS."
840 END
850 PRINT "□":REM  REPLACE AS ABOVE
860 PRINT:PRINT "I WIN!"
870 PRINT:PRINT "IN";C;"ROUNDS"
880 END
900 A(X(W)+A,Y(W)+B)=1
910 A(X(W),Y(W))=0
920 X(W)=X(W)+A
930 Y(W)=Y(W)+B
940 GOTO 200
1000 PRINT "□":REM  REPLACE AS ABOVE
1010 PRINT "              111111"
1020 PRINT "    1234567890123 45"
1030 FOR A=1 TO 15
1040 IF A<10 THEN PRINT " ";A;
1050 IF A>9  THEN PRINT A;
1060 FOR B=1 TO 15
1070 IF A(B,A)=1 THEN PRINT "#";:GOTO 1100
1080 IF A(B,A)=2 THEN PRINT "+";:GOTO 1100
1090 PRINT ".";
1100 NEXT B
1110 PRINT
1120 NEXT A
1130 RETURN
```



19

SEVEN CHURCHES

The second and third chapters of Revelation consist of short letters to seven churches in Asia Minor. The church at Ephesus is criticized for being sound in doctrine but weak in love. The church in Smyrna is poor but rich. Heretics abound at Pergamos, and a false prophetess is misleading the members of the church in Thyatira. The church at Sardis is nearly dead. Philadelphia receives praise for being loyal, and the church at Laodicea is distasteful because it is neither hot nor cold.

These seven churches become the subject of one of our more

one. If computers could complain of fatigue you would get some indication of it after this game is run. There are no time-delay loops in the setting up of the grid. The computer is actually taking all that time to figure things out.

You will need about 6K of usable memory for this one. If you omit certain aspects of the game you might be able to crunch it into 4K, but we have not tried to do so. The DIMensioning features of lines 106 and 108 take a big BYTE out of your memory.

Let's try to understand the working of the game first, and then we can go through the program together and look at some of the important areas.

The computer will select five of the seven churches for each game. It will place them on a 12 by 12 grid. But here is the catch! It will not only place them across and down, crossword-puzzle style, it will also place them diagonally, in *both* directions. There are eight possible directions for each city name, and they may share common letters at any point. Your computer will have a terrific challenge placing these words on the grid. Some day you can try to figure the statistical possibilities for five words in eight directions on a 12 by 12 grid. The process can take the computer several minutes, or it can be done in ten or fifteen seconds. No two games are alike. Line 180 keeps the player from thinking the computer has died. Line 665 gives an entertaining check on the computer's progress through its tribulations. The reason two churches are left in the DATA is to give the computer some alternatives. If it tries a church in all directions and it just won't fit in that particular game, it will throw it away and try one of the remaining two. It will make hundreds of checks for you while you twiddle your thumbs.

If the computer gets itself into an impossible jam (we have never seen it do that, but felt the eventuality should be provided for), line 690 will let it give up. Just type RUN and wish it better luck next time.

Once the names of the churches are hidden on the grid, the player is asked to enter the coordinates of each attempt to find them. First enter the number across (the top numbers), add a comma, and then the number down (the side numbers). If you are lucky enough to actually hit a letter, the computer will position it on the screen. If you hit an empty space it will place an asterisk there to remind both you and the computer that it has been tried.

When you discover your first letter, the challenge has just begun. There are now eight, count them, eight possibilities for the

name of another church! Since the names may be spelled both backwards and upside down, and since a letter may be shared by two names, you have got a bundle of problems on your hands.

You will need to use the empty opening lines to add some kind of instructions to your players. It would be nice to provide a list of the names of the seven churches and an explanation of how to enter the coordinates. This part is left entirely up to you.

The spelling of the names of the churches has been taken from the King James Version. The Good News Bible spells *Pergamos* as *Pergamum*. Select the version most familiar to your players.

Notice that the DATA lines come early in this game. Obviously, you can put them at either end.

Line 104 is critical. You must get it absolutely right. Let someone else help you proofread it after you enter it. This is the code for the eight directions of the grid. The order and the placement of commas and minus signs is unforgiving. It is hard to tell if you have made a mistake. Check and double-check.

Lines 106 and 108 set up the many variables. MS(12,12) produces the grid itself. Again, accuracy here is exceedingly important. This is a good place to start looking for errors if your game refuses to come to life.

Line 210 randomly picks a church from among the seven in DATA. Line 310 selects one of eight directions in which to throw it. The computer is given 100 chances in line 400 to hide successfully the church it has chosen in a single direction. Yes, there are 144 options, but if it hasn't found what it is looking for in the first 100 attempts, the odds are extremely unfavorable and any remaining labor is likely to be a futile wasting of time. Even with this limit, it can actually make eight hundred checks before returning to the DATA for another name. The grid locations are randomly selected by line 420. The loops starting at lines 700 and 760 clear the flags on the grid in the event of a failure, allowing new passes with a new word.

The loops from 800 to 840 clear M$. All information is thrown away so that the answers are not revealed when the grid is printed. Answers are stored in LT$ by PX and PY coordinates. (If we keep this up, we will be learning something about computers as well as about the Bible!)

A nice touch is included in line 845. The variable *D* is recycled. Since it is no longer needed for the random direction job it was assigned in line 310, it can now be reset at zero and used to keep

The main grid display is contained in lines 855 through 940. By now you should be familiar with our technique of numbering columns in 855 and 857. If you do your spaces right you will get ten, eleven, and twelve.

Line 955 was added after some players accidentally pressed the period instead of the comma to separate the coordinate entries. By forcing all entries into INTegers, the decimal point is ignored and the computer waits patiently for the second coordinate. Without this line the game was not goof proof.

The correct answer is sought by lines 1000 through 1040.

Line 1053 subtracts five points for each miss, using the recycled *D* mentioned above. I believe you could raise that number to ten or more points off and have an exciting version of the game. Your choice. In line 1113 we add ten points for a letter discovered, and in line 1167 twenty-five points are added for a word completed. Line 1166 keeps a count of the churches remaining to be finished. A loop in lines 1200 through 1220 searches for completed church names.

The final display is a work of art. Lines 1330 through 1390 remove all the dots and asterisks and put nothing but the bare names of the churches on the screen. The effect is simply beautiful. And take a good look. There is virtually no chance that you will ever see that particular display again. For all practical purposes, every game is different.

Seven Churches

```
10 REM  "SEVEN CHURCHES"
100 DATA EPHESUS,SMYRNA,PERGAMOS,THYATIRA
102 DATA SARDIS,PHILADELPHIA,LAODICEA
104 DATA 1,0,1,1,0,1,-1,1,-1,0,-1,-1,0,-1,+1,-1
106 DIM W$(7),L$(12),M$(12,12),DX(8),DY(8),DF(8),WF(7),LC(2)
108 DIM LT$(12,5),PX(12,5),PY(12,5),WD$(5)
110 FOR L=1 TO 7
120 READ W$(L)
130 NEXT L
140 FOR L=1 TO 8
150 READ DX(L),DY(L)
160 NEXT L
170 PRINT "□":REM  REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
180 PRINT "I AM NOW PLACING THE CHURCHES ON THE GRID.
190 PRINT
200 FOR WL=1 TO 7
210 W=INT(RND(1)*7)+1
220 IF WF(W)=1 THEN 210
230 WF(W)=1
240 LV=LEN(W$(W))
250 FOR L=1 TO LV
260 L$(L)=MID$(W$(W),L,1)
270 NEXT L
300 FOR DL=1 TO 8
310 D=INT(RND(1)*8)+1
320 IF DF(D)=1 THEN 310
330 DF(D)=1
400 FOR PL=1 TO 100
410 FOR L=1 TO 2
420 LC(L)=INT(RND(1)*12)+1
430 NEXT L
440 CK$=M$(LC(1),LC(2))
450 IF CK$="." THEN 410
460 IF CK$<>"" AND CK$<> L$(1) THEN 410
480 X=LC(1)
490 Y=LC(2)
500 FOR L=2 TO LV
510 X=X+DX(D)
520 Y=Y+DY(D)
525 IF X>12 OR X<1 OR Y>12 OR Y<1 THEN 550
530 CK$=M$(X,Y)
540 IF CK$="." OR CK$="" OR CK$=L$(L) THEN 590
550 IF M$(LC(1),LC(2))="" THEN M$(LC(1),LC(2))="."
560 NEXT PL
570 GOSUB 700
580 NEXT DL
585 GOSUB 760
587 GOTO 680
590 NEXT L
```

```
595 WP=WP+1
596 WD$(WP)=W$(W)
600 FOR L=1 TO LV
605 M$(LC(1),LC(2))=L$(L)
606 LT$(L,WP)=L$(L)
607 PX (L,WP)=LC(1)
608 PY (L,WP)=LC(2)
610 LC(1)=LC(1)+DX(D)
620 LC(2)=LC(2)+DY(D)
630 NEXT L
640 GOSUB 700
650 GOSUB 760
665 PRINT:PRINT "I HAVE POSITIONED CITY #";WP
670 IF WP=5 THEN 800
680 NEXT WL
690 STOP
700 FOR X=1 TO 12
710 FOR Y=1 TO 12
720 IF M$(X,Y)="." THEN M$(X,Y)=""
730 NEXT Y
740 NEXT X
750 RETURN
760 FOR X=1 TO 8
770 DF(X)=0
780 NEXT X
790 RETURN
800 FOR Y=1 TO 12
810 FOR X=1 TO 12
820 M$(X,Y)=""
830 NEXT X
840 NEXT Y
845 D=0
850 PRINT "□":REM  REPLACE AS ABOVE.
851 IF D>PS THEN PS=D
852 PRINT "CHURCHES LEFT:";5-CT
853 PRINT "## PEAK SCORE:";PS
854 PRINT "CURRENT SCORE:";D
855 PRINT "                111"
857 PRINT "    123456789012"
860 FOR Y=1 TO 12
870 IF Y>9 THEN PRINT Y;:GOTO 890
880 PRINT " ";Y;
890 FOR X=1 TO 12
900 IF M$(X,Y)="" THEN PRINT ".";:GOTO 920
910 PRINT M$(X,Y);
920 NEXT X
930 PRINT
940 NEXT Y
950 PRINT:INPUT "COORDINATE (X,Y)";X,Y
955 X=INT(X):Y=INT(Y)
960 IF X>12 OR X<1 OR Y>12 OR Y<1 THEN PRINT "OUT OF BOUNDS.":GOTO 950
970 IF M$(X,Y)<>"" THEN PRINT "YOU HAVE ALREADY TRIED THAT LOCATION":GOTO 950
1000 FOR W=1 TO 5
1010 FOR L=1 TO LEN(WD$(W))
1020 IF X=PX(L,W) AND Y=PY(L,W) THEN 1100
1030 NEXT L
1040 NEXT W
1050 PRINT "NOTHING THERE."
1053 D=D-5
1055 GOSUB 1260
1060 M$(X,Y)="*"
1070 GOTO 850
1100 M$(X,Y)=LT$(L,W)
1110 PRINT "YOU'VE FOUND: ";LT$(L,W)
1113 D=D+10
1115 GOSUB 1260
1120 LT$(L,W)=""
1130 FOR L=1 TO LEN(WD$(W))
```

```
1150 NEXT L
1160 PRINT "YOU FOUND THE CHURCH:"
1165 PRINT "*** ";WD$(W);" ***"
1166 CT=CT+1
1167 D=D+25
1170 GOSUB 1260
1180 GOSUB 1260
1200 LT$(1,W)="*"
1210 FOR W=1 TO 5
1220 IF LT$(1,W)<>"*" THEN 850
1230 NEXT W
1240 GOTO 1300
1260 FOR T=1 TO 1000
1270 NEXT T
1280 RETURN
1300 PRINT "J":REM  REPLACE AS ABOVE.
1310 PRINT "GOOD WORK! YOU HAVE FOUND ALL THE CHURCHES."
1320 PRINT
1330 FOR Y=1 TO 12
1335 PRINT "   ";
1340 FOR X=1 TO 12
1350 IF M$(X,Y)="" OR M$(X,Y)="*" THEN PRINT " ";:GOTO 1370
1360 PRINT M$(X,Y);
1370 NEXT X
1380 PRINT
1390 NEXT Y
1400 PRINT:PRINT "FINAL SCORE";D
1410 PRINT:INPUT "TRY AGAIN";CK$
1420 IF LEFT$(CK$,1)="N" THEN PRINT "BYE.":END
1430 RUN
```

20

THE PERILS OF PAUL

Adventure games are a computer buff's staple diet. This one is based upon the true life of the Apostle Paul. Many people are not aware of the hazards, pains, and difficulties that accompanied this great man's faith in Christ. "The Perils of Paul" serves as a reminder that the "peace with God" that Paul wrote about in Romans 5:1 was peace in the midst of a storm.

Historical fact will not allow this game to run with too much uncertainty and random selection. If we are going to use it as a teaching tool, it will have to turn out approximately the same way

a player to lose. Somehow, all players must make it through to the end when Paul can say, "I have fought the good fight, I have finished my course, I have kept the faith." The only variations that seemed permissible were in a degree of score. The only fair way to do this is to allow a perceptive player to match up perils and strengths. Even here, there are only two pairs that make a direct connection:

Thorn in the flesh / God's grace is sufficient.

In prison / Praying and singing praises to God.

Since the first few uses of strengths are blindly chosen by an unidentified number, it is up to the player to catch and remember which is which. This is terribly unfair, but it seems to be the only way to make a scorable game out of historical fact. The rare player who catches on can have a lot of fun with this insight.

The game includes three kinds of scriptural material. There are five "Visions," twenty-four "Perils," and seven "Strengths." Each one of the visions and strengths is displayed on the screen with the reference to the Bible. The perils are taken from Paul's own catalog of them in 2 Corinthians 11:23–28, with the "thorn in the flesh" of 2 Corinthians 12:7 thrown in for good measure.

This game will not excite adventure game experts, but it will go a long way toward teaching something about the life of Paul. Instead of the hours required to play other computer adventure games, "The Perils of Paul" takes an average of only six minutes.

If you must, you can add your own touch of high adventure to it, but be careful not to distort the beautiful life this game is intended to honor. The object behind its design, remember, is storytelling.

Everything down to line 370 is introduction. The player is asked to do nothing but be a part of the setting of the scene. The use of a time-delay loop in the subroutine at line 5000 gives a rhythmic pulse that is exciting. The loop that cycles between lines 190 and 240 produces an impressive, electric flashing of the words, "Saul, Saul! Why do you persecute me?" It is as close as a computer can come to the Damascus Road experience. If you have a voice synthesizer, make good use of it here. The double reference to the time-delay subroutine in lines 180 and 410 is important. Don't leave it out. It is a part of the drama.

After we observed a number of first-time players of this game, it became apparent that the one thing that needs to be explained is that after the first "Y or N" INPUT, only the return/enter key needs to be pressed to continue. You can do this by adding a PRINT

statement above the INPUT or by simply telling the players about it.

This game has many possibilities for creative application in teaching. If you are conducting a class on the life of Paul, the few extra minutes it takes to play the game could be a delightful "extra" that will leave a lasting impression. It is a new experiment in computer storytelling that involves the participant in some elemental decision making. When that is going on, it is not likely that the student will stare out the window and pray for the hour to pass.

The Perils of Paul

```
10 REM  "THE PERILS OF PAUL"
100 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
110 PRINT:PRINT:PRINT "YOUR NAME IS SAUL."
120 GOSUB 5000
130 PRINT:PRINT "YOU ARE A PHARISEE."
140 GOSUB 5000
150 PRINT:PRINT "YOU ARE ON YOUR WAY TO DAMASCUS."
160 GOSUB 5000
170 PRINT:PRINT "YOU ARE PERSECUTING CHRISTIANS."
180 GOSUB 5000:GOSUB 5000
190 FOR F=1 TO 150
200 PRINT "□":REM  REPLACE AS ABOVE.
210 PRINT:PRINT:PRINT:PRINT:PRINT "SAUL, SAUL!"
220 PRINT:PRINT:PRINT "WHY DO YOU PERSECUTE ME?"
230 PRINT:PRINT "          -ACTS 9:4"
240 NEXT F
250 PRINT "□":REM  REPLACE AS ABOVE.
260 GOSUB 5000
270 PRINT:PRINT "YOU ARE CONVERTED."
280 GOSUB 5000
290 PRINT:PRINT "YOU BECOME A CHRISTIAN."
300 GOSUB 5000
310 PRINT:PRINT "YOU WILL SERVE CHRIST."
320 GOSUB 5000
330 PRINT:PRINT "IT WILL COST YOU."
340 GOSUB 5000
350 PRINT:PRINT "IT WILL BE A JOY."
360 GOSUB 5000:GOSUB 5000
370 PRINT "□":REM  REPLACE AS ABOVE.
380 PRINT:PRINT:PRINT:PRINT:PRINT "NOW PLAY. . ."
390 GOSUB 5000
400 PRINT:PRINT:PRINT "THE PERILS OF PAUL!"
410 GOSUB 5000:GOSUB 5000
420 PRINT "□":REM  REPLACE AS ABOVE.
430 PRINT:PRINT "PERILS WILL BE THROWN"
440 PRINT "IN YOUR PATH."
450 GOSUB 5000
460 PRINT:PRINT "THEY COST YOU POINTS."
470 GOSUB 5000
480 PRINT:PRINT "YOU MUST CHOOSE WHICH OF 7 STRENGTHS"
490 PRINT "WILL HELP THE MOST."
500 GOSUB 5000
510 PRINT:PRINT "THEY ADD POINTS."
520 GOSUB 5000
530 PRINT:PRINT "VISIONS COME AT UNEXPECTED TIMES AND ADD 100 POINTS."
540 GOSUB 5000:GOSUB 5000
545 PRINT:PRINT "HIT RETURN WHEN READY."
550 INPUT X
```

```
580 PRINT:PRINT "COME OVER TO MACEDONIA AND HELP US."
590 PRINT:PRINT "          -ACTS 16:9"
600 GOSUB 5000
610 PRINT:PRINT "WILL YOU GO?"
620 INPUT "Y OR N";A$
630 IF A$="N" THEN PRINT "YOU ARE A POOR PAUL!  HE WENT!":GOSUB 5000:GOTO 620
640 PRINT:PRINT "GOOD! BUT WATCH OUT..."
650 S=100:V=0
660 GOSUB 5000
670 RESTORE
680 P=INT(RND(1)*24)+1
690 FOR Q=1 TO P
700 READ P$
710 NEXT Q
720 S=S-20
730 GOSUB 5050
740 GOSUB 3000
750 V=V+1
760 IF V=5 THEN 1000
770 IF V=9 THEN 1100
780 IF V=14 THEN 1200
790 IF V=19 THEN 1300
800 IF V=24 THEN 2000
810 GOTO 670
1000 GOSUB 5300
1010 PRINT:PRINT "CAUGHT UP TO THIRD HEAVEN"
1020 PRINT:PRINT "       -2 COR. 12:2"
1030 GOSUB 5000
1040 PRINT:PRINT "REFRESHED AND INSPIRED. . .GO ON. . ."
1050 PRINT:PRINT
1060 INPUT "ARE YOU READY";A
1070 GOTO 670
1100 GOSUB 5300
1110 PRINT:PRINT "GET QUICKLY OUT OF JERUSALEM."
1120 PRINT:PRINT "THEY WILL NOT LISTEN TO YOU."
1130 PRINT:PRINT "      -ACTS 22:18"
1140 GOSUB 5000
1150 PRINT:PRINT "YOU ARE IN DANGER!"
1160 PRINT:PRINT "WHY DON'T YOU"
1170 INPUT "RUN";A
1180 GOTO 670
1200 GOSUB 5300
1210 PRINT:PRINT "YOU MUST BEAR WITNESS ALSO AT ROME."
1220 PRINT:PRINT "      -ACTS 23:11"
1230 GOSUB 5000
1240 PRINT:PRINT "YOU ARE NEEDED..."
1250 INPUT "WHY WAIT";A
1260 GOTO 670
1300 GOSUB 5300
1310 PRINT:PRINT "AN ANGEL SAYS, 'FEAR NOT, PAUL. YOU MUST BE"
1320 PRINT "BROUGHT BEFORE CAESAR.'"
1330 PRINT:PRINT "    -ACTS 27:23-24"
1340 GOSUB 5000
1350 PRINT:PRINT "ARE YOU WILLING TO"
1360 INPUT "CONTINUE";A
1370 GOTO 670
2000 PRINT "J":REM  REPLACE AS ABOVE
2010 PRINT:PRINT:PRINT:PRINT "I HAVE FOUGHT THE GOOD FIGHT,"
2020 GOSUB 5000
2030 PRINT:PRINT "I HAVE FINISHED MY COURSE,"
2040 GOSUB 5000
2050 PRINT:PRINT "I HAVE KEPT THE FAITH."
2060 GOSUB 5000
2070 PRINT:PRINT "  2 TIMOTHY 4:7"
2080 GOSUB 5000
2090 PRINT:PRINT:PRINT "FINAL SCORE:";S
2100 PRINT:PRINT:PRINT
2110 END
```

```
3020 ON Z GOTO 3030, 3040, 3050, 3060, 3070
3030 PRINT:PRINT "WHAT NEXT?":RETURN
3040 PRINT:PRINT "THERE ARE MORE HAZARDS":RETURN
3050 PRINT:PRINT "LOOK OUT! HERE IS":RETURN
3060 PRINT:PRINT "DON'T QUIT NOW, YOU'VE GOT":RETURN
3070 PRINT:PRINT "YOUR ENDURANCE IS BEING TESTED":RETURN
5000 FOR T=1 TO 1500:NEXT T
5010 RETURN
5050 PRINT:PRINT "A PERIL!"
5060 PRINT:PRINT "YOU ARE ";P$
5070 FOR T=1 TO 1200:NEXT T
5080 PRINT:PRINT "SELECT A STRENGTH"
5090 PRINT:PRINT
5100 INPUT "1 TO 7";C
5105 IF C>7 THEN 5100
5110 ON C GOTO 5120, 5130, 5140, 5150, 5160, 5170, 5180
5120 PRINT:PRINT "PRAYING & SINGING PRAISES TO GOD  -ACTS 16:25":GOTO 5190
5130 PRINT:PRINT "PATIENCE  -2 COR. 12:12":GOTO 5190
5140 PRINT:PRINT "COURAGE  -ACTS 16:36,37":GOTO 5190
5150 PRINT:PRINT "TENTMAKING  -ACTS 20:33,34":GOTO 5190
5160 PRINT:PRINT "STEADFASTNESS  -ACTS 20:24":GOTO 5190
5170 PRINT:PRINT "GOD'S GRACE IS SUFFICIENT  2 COR. 12:9":GOTO 5190
5180 PRINT:PRINT "CHEERFULNESS  PHIL. 4:4":GOTO 5190
5190 S=S+10
5200 IF C=6 AND P=24 THEN S=S+50
5210 IF C=1 AND P=1 THEN S=S+50
5220 FOR T=1 TO 1400: NEXT T
5230 PRINT:PRINT "CURRENT SCORE:"S
5240 RETURN
5300 PRINT:PRINT:PRINT "****** VISION *******"
5310 S=S+100
5320 RETURN
5990 REM  THE LIST BELOW IS TAKEN FROM 2 COR. 11:23-28
6000 DATA "IN PRISON","WHIPPED!","NEAR DEATH","LASHED 39 TIMES"
6010 DATA "BEATEN WITH RODS","HIT WITH STONES","SHIPWRECKED"
6020 DATA "ADRIFT AT SEA","IN A FLOOD","ROBBED BY THIEVES"
6030 DATA "FOUGHT BY ENEMIES","IN PERIL IN CITIES"
6040 DATA "IN PERIL IN WILDERNESS","IN PERIL AT SEA"
6050 DATA "HURT BY FALSE FRIENDS","WEARY","IN PAIN","SLEEPY"
6060 DATA "HUNGRY","THIRSTY","COLD","POORLY CLOTHED"
6070 DATA "CONCERNED FOR THE CHURCHES"
6079 REM  THE LAST IS FROM 2 COR. 12:7
6080 DATA "SICK WITH A THORN IN THE FLESH"
```

21

BIBLE PAIRS

This game allows a new level of participation by the computer. It actually ''learns'' things from the player's moves. This means that the computer plays a gradually more intelligent game. It could cheat, but a part of the computer has been set aside to prevent this. There are actually three entities involved in each game: the human player, the computer as player, and the computer as referee.

The substance of the game is based on pairs of people and things mentioned in the Bible. A connecting description and number make each pair clear to the player. *Alpha* goes with *Omega* because both

At the beginning of play, the computer randomly selects several halves from the DATA for both itself and the human challenger. It keeps its own a secret and displays the player's on the screen. Again, the playing part of the computer can't "see" the screen. The player asks the computer for one of its halves in order to complete a pair. If the playing part of the computer has the requested half, it will surrender it. If it does not, the computer referee will randomly choose another half from DATA. This is displayed on the screen along with the previous collection. If the player has a pair, it can be set aside and another turn is granted. If there is no pair, then the computer has a chance to request a half from the human player.

This is where the game gets interesting. Now both human and electronic brains know something about each other. The computer will *remember* what you have requested in the past. It can begin to deduce what halves you are holding. The object is to get rid of all your halves. As in golf, low scores are best.

Let it be emphasized that the playing section of the computer does not know what halves the human player possesses at the beginning of the game. It could, but we have been scrupulous about this. It can infer only that if you ask for one half, you have the other half. This information is stored in IT$ (for intelligence) in line 583. If the computer attaches a +, it thinks you have the other half. If it attaches a −, it thinks you don't.

The other function of the computer is the prevention of deception. If the computer as player requests a particular half and the human player denies having it, the computer as referee objects. This isolated part of the computer knows everything that is going on. If the human player makes an honest mistake, the mediator will gently point it out. You can play a shrewd game, but deception is impossible. In essence, this program gives you two computers in one. CM$ is the computer memory and PM$ is the player memory.

Your brain is on its own. Enjoy!

Bible Pairs

```
10 REM  "BIBLE PAIRS"
100 DATA "GREEK LETTERS",ALPHA,OMEGA
101 DATA "HAIRCUT",SAMSON,DELILAH
102 DATA "FIRST COUPLE",ADAM,EVE
103 DATA "OVERLAPPING PROPHETS",ELIJAH,ELISHA
104 DATA "FIRST BLOOD",CAIN,ABEL
105 DATA "IN-LAWS",RUTH,NAOMI
106 DATA "HEBREW PARENTS",ABRAHAM,SARAH
107 DATA "WHATSITS",GOG,MAGOG
108 DATA "NAZARETH COUPLE",JOSEPH,MARY
109 DATA "SOLOMON'S PARENTS",DAVID,BATHSHEBA
110 DATA "MISSIONARIES",PAUL,SILAS
111 DATA "MOTHER & GRANDMOTHER",LOIS,EUNICE
112 DATA "LIE DETECTOR",ANANIAS,SAPHIRA
113 DATA "BAPTISTS?",ZECHARIAH,ELIZABETH
114 DATA "BROTHERS",JACOB,ESAU
115 DATA "OLD & YOUNG",ELI,SAMUEL
116 DATA "TWO VOLUMES",LUKE,ACTS
117 DATA "PRIESTLY GADGETS",URIM,THUMMIM
118 DATA "FOR THE LORD",TITHES,OFFERINGS
120 DIM PM$(38),CM$(38),DH$(38),IT$(19),DC$(19)
130 FOR L=2 TO 38 STEP 2
140 READ DC$(L/2),DH$(L-1),DH$(L)
150 NEXT L
160 DEF FNR(R)=INT(RND(1)*38)+1
170 FOR L=1 TO 5
180 R=FNR(R)
190 IF DH$(R)="" THEN 180
200 PM$(R)=DH$(R):DH$(R)=""
210 NEXT L
220 FOR L=1 TO 5
230 R=FNR(R)
240 IF DH$(R)="" THEN 230
250 CM$(R)=DH$(R):DH$(R)=""
270 NEXT L
280 GOSUB 2030
370 IF B>5 THEN GOSUB 2000
400 FOR L=1 TO 38
401 IF PM$(L)<>"" THEN 409
402 NEXT L
403 PRINT "YOU HAVE NO MORE PAIRS, YOU WIN."
404 GOTO 1300
409 PRINT "DO YOU HAVE ANY PAIRS"
410 INPUT A$
420 IF LEFT$(A$,1)="N" THEN 500
430 PRINT "ENTER THE PAIR NUMBER"
440 INPUT A
450 IF A<1 OR A>19 THEN PRINT "THAT IS IMPOSSIBLE.":GOSUB 3000:GOTO 430
455 B=A*2:C=A*2-1
460 IF PM$(B)=""OR PM$(C)="" THEN PRINT"THAT'S NOT CORRECT.":GOSUB 3000:GOTO 400
470 PRINT:PRINT "PAIR #";A;"IS NOW COMPLETED, AND SET ASIDE."
```

```
480 PM$(B)="":PM$(C)=""
490 GOSUB 2050:GOTO 260
500 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
510 FOR L=2 TO 38 STEP 2
520 IF PM$(L)<>"" AND PM$(L-1)<>""THEN PRINT "YES YOU DO!":GOSUB 2000:GOTO 400
530 NEXT L
540 PRINT:PRINT "YOU MAY ASK ME FOR ONE OF MY BIBLE PAIRS."
550 PRINT:PRINT "IF I HAVE PART OR ALL OF A PAIR, I WILL GIVE IT TO YOU."
560 PRINT:PRINT "WHAT IS THE NUMBER OF THE PAIR YOU WANT"
570 INPUT A
580 IF A>19 OR A<1 THEN PRINT "NUMBERS 1-19 ONLY.":GOTO 570
583 IT$(A)="+"
585 B=A*2:C=A*2-1
590 IF CM$(B)="" AND CM$(C)="" THEN 700
600 IF CM$(B)<>"" AND CM$(C)<>""THEN PRINT "DRAT! I HAD THE WHOLE PAIR.":GOTO620
610 PRINT:PRINT "I HAVE A PART OF THAT PAIR."
620 PRINT:PRINT "IT IS NOW IN YOUR MEMORY AREA."
630 IF PM$(B)="" THEN PM$(B)=CM$(B):CM$(B)=""
640 IF PM$(C)="" THEN PM$(C)=CM$(C):CM$(C)=""
645 GOSUB 2050
650 GOTO 200
700 PRINT "□":REM  REPLACE AS ABOVE.
710 GOSUB 2070
711 PRINT "I DONT HAVE ANY ´";DC$(A);"´ SO YOU WILL HAVE TO DRAW."
712 R=FNR(R)
713 IF DH$(R)="" THEN 712
714 PM$(R)=DH$(R):DH$(R)=""
715 C=INT(R/2+.5)
720 PRINT:PRINT "YOU GET #";C;DC$(C);" :";PM$(R)
730 GOSUB 2050
731 IF C<>A THEN 800
732 PRINT:PRINT "THAT´S WHAT YOU ASKED ME FOR, SO GO AGAIN."
735 GOSUB 2050
740 GOTO 260
800 PRINT "□":REM  REPLACE AS ABOVE.
801 FOR L=1 TO 38
802 IF CM$(L)<>"" THEN 806
803 NEXT L
804 PRINT "I HAVE NO MORE PAIRS - I WIN."
805 GOTO 1300
806 CH=0
807 FOR L=1 TO 19
808 IF CM$(L*2)<>"" OR CM$(L*2-1)<>"" THEN CH=CH+1
809 NEXT L
810 PRINT "IT IS NOW MY TURN."
815 PRINT:PRINT "I HAVE";CH;"HALVES LEFT."
820 FOR L=1 TO 19
830 B=L*2:C=L*2-1
840 IF CM$(B)<>"" AND CM$(C)<>"" THEN 880
850 NEXT L
860 PRINT:PRINT "I HAVE NO PAIRS."
870 GOTO 900
880 PRINT:PRINT "I HAVE A PAIR!"
890 CM$(B)="":CM$(C)=""
900 GOSUB 3000:GOSUB 2030
905 PRINT:PRINT "NOW I WILL ASK YOU FOR A BIBLE PAIR."
910 FOR L=1 TO 19
920 IF IT$(L)<>"+" THEN 930
922 IF CM$(L*2)<>"" OR CM$(L*2-1)<>"" THEN R=L:GOTO 950
930 NEXT L
935 FOR L=1 TO 19
940 R=INT(RND(1)*19)+1
950 IF IT$(R)="-" THEN 957
955 IF CM$(R*2)="" AND CM$(R*2-1)="" THEN 957
956 GOTO 960
957 NEXT L
958 R=INT(RND(1)*19)+1
959 IF CM$(R*2)="" AND CM$(R*2-1)="" THEN 958
```

```
975 B=R*2:C=R*2-1
980 IF LEFT$(A$,1)="N" THEN 1100
1000 IF PM$(B)="" AND PM$(C)="" THEN PRINT "NO YOU DON'T.":GOSUB 2050:GOTO 960
1010 IF CM$(B)="" THEN CM$(B)=PM$(B):PM$(B)=""
1020 IF CM$(C)="" THEN CM$(C)=PM$(C):PM$(C)=""
1025 IT$(R)="-"
1027 PRINT:PRINT "#";R;"IS NOW IN MY MEMORY."
1028 GOSUB 3000
1030 GOTO 600
1100 IF PM$(B)<>"" OR PM$(C)<>"" THEN PRINT "THAT'S NOT SO.":GOSUB 2050:GOTO 960
1105 IT$(R)="-"
1110 GOSUB 2070
1115 A=R
1120 PRINT:PRINT "I AM NOW DRAWING FOR A RANDOM PIECE."
1125 GOSUB 3000
1130 R=FNR(R)
1140 IF DH$(R)="" THEN 1130
1150 CM$(R)=DH$(R):DH$(R)=""
1160 IF A=R THEN 1200
1170 PRINT:PRINT "I DIDN'T GET THE ONE I WANTED - IT'S YOUR TURN."
1180 GOSUB 2050
1190 GOTO 280
1200 PRINT "THAT'S THE ONE I ASKED YOU FOR!  I GET TO GO AGAIN."
1210 GOSUB 2050
1220 GOTO 600
1300 GOSUB 3000
1305 PRINT "□":REM  REPLACE AS ABOVE.
1310 PRINT "FINAL SCORE:"
1320 PRINT:PRINT "COMPUTER";CH
1330 PRINT:PRINT "PERSON  ";B
1340 PRINT:PRINT "PLAY AGAIN"
1350 INPUT A$
1360 IF LEFT$(A$,1)="Y" THEN RUN
1370 PRINT:PRINT "GOOD-BYE."
1999 END
2000 PRINT:PRINT "WOULD YOU LIKE TO SEE THEM AGAIN"
2010 INPUT A$
2020 IF LEFT$(A$,1)="N" THEN RETURN
2030 PRINT "□":REM  REPLACE AS ABOVE.
2031 B=0
2032 FOR L=1 TO 38
2033 IF PM$(L)="" THEN 2039
2034 FOR T=1 TO 200:NEXT T
2035 B=B+1
2036 C=INT(L/2+.5)
2037 PRINT "#";C;DC$(C);": ";PM$(L)
2038 PRINT
2039 NEXT L
2040 RETURN
2050 PRINT:INPUT "READY TO CONTINUE";A$
2060 RETURN
2070 FOR L=1 TO 38
2080 IF DH$(L)<>"" THEN RETURN
2090 NEXT L
2100 PRINT "DATA EMPTY.":GOTO 1300
3000 FOR T=1 TO 1000
3010 NEXT T
3020 RETURN
```

Earlier in these games I urged you to use your imagination and do some computer doodling. Room for improvement has been provided in every program. Here is an example of how to use those

Using loops and variable TAB statements, I was able to animate things on the screen. A lonely *O* "walks" across the top of the screen and finds a mate. The pair returns to the right side and rests. An *X* then does the same thing. You can change their speed or use any other characters you like.

Then, just for fun, I animated an entire word. *Bible* crosses the screen and picks up *Pairs*. Line 90 causes these words to space themselves properly for a final title.

Study this program and then create something better of your own.

Bible Pairs (animated introduction)

```
1 REM  ANIMATED INTRODUCTION FOR "BIBLE PAIRS"
12 FOR R=0 TO 21
14 PRINT "□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
16 PRINT:PRINT:PRINT TAB(R) "O";
18 FOR T=1 TO 100:NEXT T
20 NEXT R
22 FOR L=19 TO 0 STEP-1
24 PRINT "□":REM  REPLACE AS ABOVE.
26 PRINT:PRINT:PRINT TAB(L) "O-O"
28 FOR T=1 TO 100:NEXT T
30 NEXT L
32 FOR R=0 TO 21
34 PRINT "□":REM  REPLACE AS ABOVE.
36 PRINT:PRINT:PRINT "O-O"
38 PRINT:PRINT TAB(R) "X";
40 FOR T=1 TO 100:NEXT T
42 NEXT R
44 FOR L=19 TO 0 STEP-1
46 PRINT"□":REM  REPLACE AS ABOVE.
48 PRINT:PRINT:PRINT "O-O":PRINT
50 PRINT TAB(L) "X-X"
52 FOR T=1 TO 100:NEXT T
54 NEXT L
56 FOR R=0 TO 16
58 PRINT "□":REM  REPLACE AS ABOVE.
60 PRINT:PRINT:PRINT "O-O"
62 PRINT:PRINT "X-X"
64 PRINT:PRINT TAB(R) "BIBLE>"
66 FOR T=1 TO 25:NEXT T
68 NEXT R
70 FOR L=10 TO 4 STEP-1
72 PRINT "□":REM  REPLACE AS ABOVE.
74 PRINT:PRINT:PRINT "O-O"
76 PRINT:PRINT "X-X"
78 PRINT:PRINT TAB(L) "BIBLE><PAIRS"
80 FOR T=1 TO 20:NEXT T
82 NEXT L
84 PRINT "□":REM  REPLACE AS ABOVE.
86 PRINT:PRINT:PRINT "O-O"
88 PRINT:PRINT "X-X"
90 PRINT:PRINT TAB(4) "BIBLE >< PAIRS"
92 FOR T=1 TO 900:NEXT T
```

22

REVERSE WHICH BOOK?

This one is for the dedicated few who enjoy a complicated program! The amazing things that happen when the game is played make the effort worthwhile. The test version could not be accommodated by less than 16K. Some refinements have reduced its needs to the capacity of most personal computers. With crunching, you can get it into less than 6K. If you are clever, you can make it even more compact than that. Our aim was to produce a program that could be copied easily, with few mistakes.

In this game, the *player* picks a book of the Bible, and the

book with fewer than ten questions. In fact, it can sometimes do it with four questions. The average is six.

The effect is spooky. The computer appears to be thinking. It seems to be a living being. One young woman was so awestruck, she actually turned her head away from the computer and whispered her secret choice.

The computer narrows down the possibilities by some cold logic. It quickly follows each branch to a conclusion. The intelligence, however, is the programmer's. If you could remember the sequences and the books in the final random pods, you could do this one without the aid of a computer.

The question in line 400 cuts the possibilities way down. Each question that follows continues to narrow down the options until only a few remain. A random generator in line 7030 suggests each possibility in an unpredictable sequence.

The catch is in the quality of the player's answers. If the victim doesn't know whether or not Galatians is Pauline, such ignorance will soon be exposed. Lines 7510–7600 do the taunting. If the questions are answered correctly, the computer always wins. It is important, therefore, to understand that the logic follows the traditional categories of biblical literature. It knows nothing of the debate about the authorship of Hebrews.

The game opens with an animated title. The word *Reverse* jumps from one end of the line to the other six times. The remainder of the introduction should be clear to you by now. Note the space-saving strings in lines 340–380.

There is nothing mysterious about the body of the program. The computer is fairly relaxed until a branch of logic has been followed to its limit. An example of the change that takes place at that point can be seen in line 2320. The player is asked if the secret book is a Gospel. If it is, line 2370 assigns unique values to the variables *X* and *Y*. *X* is easy. There are four Gospels. *Y* tells the computer where to start looking in the DATA. In this case, it is with item twenty-two.

Take a close look at the DATA. These lines are quite unlike the DATA lines in the original "Which Book?" The order has nothing to do with the order of the books in the Bible. It is strictly determined by the logical sequence of the game. Make a mistake after line 9000 and you are in for trouble. Notice the quotation marks around any book with a space in its title. Don't miss the question mark after Acts. Revelation rates an entire sentence. Copy everything *exactly*

The final selection for each line of reasoning is made by lines 7000–7150. The only exception is Revelation, which is reached by a direct route with no alternatives.

Reverse Which Book?

```
10 REM  "REVERSE WHICH BOOK?"
20 REM  BE SURE TO USE YOUR CLEAR SCREEN COMMAND IN LINES 150 & 170.
100 DIM B$(66)
110 FOR B=1 TO 66
120 READ B$(B)
130 NEXT B
140 FOR L=1 TO 3
150 PRINT "□":PRINT:PRINT:PRINT "**REVERSE WHICH BOOK?"
160 GOSUB 8000
170 PRINT "□":PRINT:PRINT:PRINT "WHICH BOOK? REVERSE**"
180 GOSUB 8000
190 NEXT L
200 PRINT:PRINT:PRINT "I CAN GUESS ANY BOOK  OF THE BIBLE"
210 PRINT "WITH LESS THAN TEN QUESTIONS."
220 GOSUB 8020
230 PRINT:PRINT "BUT YOU MUST KNOW THE CORRECT ANSWERS.":PRINT:PRINT
240 GOSUB 8000
250 INPUT "ARE YOU WILLING";A$
260 PRINT "□":REM  REPLACE AS ABOVE.
270 PRINT:PRINT:PRINT "THINK OF A BOOK OF THE BIBLE."
280 GOSUB 8000
290 PRINT:PRINT "TELL SOMEONE OR WRITE IT ON PAPER."
300 GOSUB 8020
310 PRINT:PRINT
320 INPUT "GOT IT";A$
330 PRINT "□":REM  REPLACE AS ABOVE.
340 I$="IS IT ":D$="DOES IT "
350 N$="I HAVE NARROWED IT TO "
360 P$=" POSSIBILITIES."
370 Q$=">> MY QUESTION #"
380 M$="THEN IT MUST BE ":Q=1
390 PRINT:PRINT Q$;Q
400 PRINT:PRINT:PRINT "WHICH TESTAMENT?":PRINT
410 INPUT "OLD OR NEW";A$
420 IF A$="OLD" THEN 450
430 IF A$="NEW" THEN 2000
440 GOTO 410
450 Z=39
460 GOSUB 6500
470 PRINT:PRINT I$;"A PROPHET?"
480 INPUT "Y OR N";A$
490 IF A$="Y" THEN 520
500 IF A$="N" THEN 860
510 GOTO 480
520 Z=17:GOSUB 6500
530 PRINT:PRINT
540 INPUT "MAJOR OR MINOR";A$
550 IF A$="MAJOR" THEN 580
560 IF A$="MINOR" THEN 600
570 GOTO 540
```

```
580 X=5:Y=62
590 GOTO 7000
600 Z=12:GOSUB 6500
610 PRINT:PRINT D$;"HAVE 5 LETTERS?"
620 INPUT "Y OR N";A$
630 IF A$="Y" THEN 660
640 IF A$="N" THEN 680
650 GOTO 620
660 X=4:Y=58
670 GOTO 7000
680 Z=8:GOSUB 6500
690 PRINT:PRINT D$;"START WITH H?"
700 INPUT "Y OR N";A$
710 IF A$="Y" THEN 740
720 IF A$="N" THEN 760
730 GOTO 700
740 X=2:Y=50
750 GOTO 7000
760 Z=6:GOSUB 6500
770 PRINT:PRINT D$;"START WITH A VOWEL?"
780 INPUT "Y OR N";A$
790 IF A$="Y" THEN 820
800 IF A$="N" THEN 840
810 GOTO 780
820 X=2:Y=44
830 GOTO 7000
840 X=4:Y=46
850 GOTO 7000
860 Z=22:GOSUB 6500
870 PRINT:PRINT I$;"HISTORY, AS OPPOSED TO PENTATEUCH OR POETRY?"
880 INPUT "Y OR N";A$
890 IF A$="Y" THEN 920
900 IF A$="N" THEN 1180
910 GOTO 880
920 Z=12:GOSUB 6500
930 PRINT:PRINT D$;"HAVE 2 PARTS?"
940 INPUT "Y OR N";A$
950 IF A$="Y" THEN 980
960 IF A$="N" THEN 1000
970 GOTO 940
980 X=6:Y=52
990 GOTO 7000
1000 Z=6:GOSUB 6500
1010 PRINT:PRINT D$;"TELL OF CONQUEST & EARLY SETTLEMENT?"
1020 INPUT "Y OR N";A$
1030 IF A$="Y" THEN 1060
1040 IF A$="N" THEN 1080
1050 GOTO 1020
1060 X=2:Y=32
1070 GOTO 7000
1080 Z=4:GOSUB 6500
1090 PRINT:PRINT I$;"ABOUT A WOMAN?"
1100 INPUT "Y OR N";A$
1110 IF A$="Y" THEN 1140
1120 IF A$="N" THEN 1160
1140 X=2:Y=28
1150 GOTO 7000
1160 X=2:Y=30
1170 GOTO 7000
1180 Z=10:GOSUB 6500
1190 PRINT:PRINT I$;"IN THE PENTATEUCH?"
1200 INPUT "Y OR N";A$
1210 IF A$="Y" THEN 1240
1220 IF A$="N" THEN 1260
1230 GOTO 1200
1240 X=5:Y=34
1250 GOTO 7000
1260 X=5:Y=39
```

```
2010 GOSUB 6500
2020 PRINT:PRINT I$;"AN EPISTLE?"
2030 INPUT "Y OR N";A$
2040 IF A$="Y" THEN 2070
2050 IF A$="N" THEN 2300
2060 GOTO 2030
2070 Z=21
2080 GOSUB 6500
2090 PRINT:PRINT I$;"PAULINE?"
2100 INPUT "Y OR N";A$
2110 IF A$="Y" THEN 2140
2120 IF A$="N" THEN 2390
2130 GOTO 2100
2140 Z=14
2150 GOSUB 6500
2160 PRINT:PRINT D$;"HAVE TWO PARTS?"
2170 INPUT "Y OR N";A$
2180 IF A$="Y" THEN 2210
2190 IF A$="N" THEN 2480
2200 GOTO 2170
2210 Z=6
2220 GOSUB 6500
2230 PRINT:PRINT D$;"START WITH T?"
2240 INPUT "Y OR N";A$
2250 IF A$="Y" THEN 2280
2260 IF A$="N" THEN 2590
2270 GOTO 2240
2280 X=4:Y=9
2290 GOTO 7000
2300 Z=6:GOSUB 6500
2320 PRINT:PRINT I$;"A GOSPEL?"
2330 INPUT "Y OR N";A$
2340 IF A$="Y" THEN 2370
2350 IF A$="N" THEN 2610
2360 GOTO 2330
2370 X=4:Y=22
2380 GOTO 7000
2390 Z=7:GOSUB 6500
2410 PRINT:PRINT D$;"HAVE MORE THAN 1 PART?"
2420 INPUT "Y OR N";A$
2430 IF A$="Y" THEN 2460
2440 IF A$="N" THEN 2720
2450 GOTO 2420
2460 X=5:Y=15
2470 GOTO 7000
2480 Z=8:GOSUB 6500
2500 PRINT:PRINT D$;"USE A PLACE NAME?"
2510 INPUT "Y OR N";A$
2520 IF A$="Y" THEN 2550
2530 IF A$="N" THEN 2570
2540 GOTO 2510
2550 X=5:Y=4
2560 GOTO 7000
2570 X=3:Y=1
2580 GOTO 7000
2590 X=2:Y=13
2600 GOTO 7000
2610 Z=2:GOSUB 6500
2615 N=26
2620 PRINT:PRINT I$;B$(N)
2630 INPUT "Y OR N";A$
2640 IF A$="Y" THEN 7500
2650 IF A$="N" THEN 2670
2660 GOTO 2630
2670 PRINT:PRINT B$(27)
2675 PRINT
2680 INPUT "RIGHT OR WRONG";A$
2690 IF A$="RIGHT" THEN 7500
```

```
2720 X=2:Y=20
2730 GOTO 7000
6500 PRINT:PRINT N$;Z;P$
6510 GOSUB 8000
6520 Q=Q+1
6530 PRINT:PRINT Q$;Q
6540 RETURN
7000 Z=X
7005 FOR L=1 TO X
7007 IF Z=1 THEN 7030
7010 PRINT:PRINT N$;Z;P$:PRINT
7030 N=INT(RND(1)*X)+Y
7040 FOR S=1 TO L
7050 IF RF(S)=N THEN 7030
7060 NEXT S
7070 RF(L)=N
7080 GOSUB 8000
7090 Q=Q+1
7095 IF Z=1 THEN I$=M$:GOTO 7110
7100 PRINT:PRINT Q$;Q
7110 PRINT:PRINT I$;B$(N);"?"
7120 INPUT "Y OR N";A$
7130 IF A$="Y" THEN 7500
7135 Z=Z-1
7140 NEXT L
7150 GOTO 7510
7500 PRINT:PRINT "SUCCESS! I DID IT WITH";Q;"QUESTIONS.":GOTO 7540
7510 PRINT:PRINT "*****   PROBLEM NOT IN MY CIRCUITS."
7520 GOSUB 8000
7530 PRINT:PRINT "** YOU GAVE INACCURATE RESPONSES. GO STUDY!":PRINT:PRINT
7540 GOSUB 8000
7550 PRINT:PRINT:PRINT "DO YOU WANT TO ATTEMPT "
7560 INPUT "ANOTHER";A$
7570 IF A$="YES" OR A$="Y" THEN RUN
7580 PRINT:PRINT
7590 IF A$="NO" OR A$="N" THEN PRINT "GOD BE WITH YOU..."
7600 END
8000 FOR T=1 TO 777:NEXT T
8010 RETURN
8020 FOR T=1 TO 2000:NEXT T
8030 RETURN
9000 DATA TITUS,PHILEMON,HEBREWS,ROMANS,GALATIANS,EPHESIANS
9010 DATA PHILIPPIANS,COLOSSIANS,"1 THESSALONIANS","2 THESSALONIANS"
9020 DATA "1 TIMOTHY","2 TIMOTHY","1 CORINTHIANS","2 CORINTHIANS"
9030 DATA "1 PETER","2 PETER","1 JOHN","2 JOHN","3 JOHN",JAMES,JUDE
9040 DATA MATTHEW,MARK,LUKE,JOHN,ACTS?,"IT'S GOT TO BE REVELATION."
9050 DATA RUTH,ESTHER,EZRA,NEHEMIAH,JOSHUA,JUDGES,GENESIS,EXODUS
9060 DATA LEVITICUS,NUMBERS,DEUTERONOMY,JOB,PSALMS,PROVERBS
9070 DATA ECCLESIASTES,"SONG OF SOLOMON",OBADIAH,AMOS,JOEL,ZEPHANIAH
9080 DATA ZECHARIAH,MALACHI,HABAKKUK,HAGGAI,"1 SAMUEL","2 SAMUEL"
9090 DATA "1 KINGS","2 KINGS","1 CHRONICLES","2 CHRONICLES"
9100 DATA HOSEA,JONAH,MICAH,NAHUM,ISAIAH,JEREMIAH
9110 DATA LAMENTATIONS,EZEKIEL,DANIEL
```

Appendix

Brief Routines Suitable for Rewards

It was suggested in the introductory chapters that you consider adding special rewards to the games in this book. If your computer's memory capacity will allow it, you can greatly increase the player's enjoyment by tacking on some sort of pat on the back for a good score. The simple routines presented here are examples intended to inspire your own imagination.

The line numbers can get you into trouble. Be sure to change them so that they will not "bump" any important lines already in your existing program. And remember that when you change line numbers, you must also go through the entire routine and look for all statements that refer to another numbered line. Those numbers must be revised according to your new sequence. This is not difficult, but accuracy is mandatory.

A. *Prize Routine*

The first routine does not take any of its subject matter from the Bible. It contains twenty-six "prizes" that bring smiles to players in Florida. You can replace them with ideas of your own invention.

The player is invited to make a blind choice by pressing any key with a letter on it. If some other key is pressed, line 1060 ignores it and makes line 1050 continue waiting patiently for a legal entry.

Take a close look at line 1050. It contains a GET statement. These have been scrupulously avoided in the main programs because a few computers are particular in their requirements regarding these. Using it here allows most players simply to press the one chosen key to set off a response. If this routine will not RUN on your computer, the simplest solution will be to change the GET to an INPUT. Then the player will have to press both a letter key and the return/enter key, but it will work. On a TRS-80, try A$ = INKEY$.

There is a gimmick familiar to computer hackers that does nothing at all in lines 1090 and 1110. That is, nothing *essential*. All they do is throw in a delaying tactic that makes the computer *appear* to be working. Printing the line, "I am searching," enhances the illusion. Actually, nothing is going on except counting to twelve hundred. Players respond favorably to this ruse. They like to imagine electronic things humming away through some complicated maze of wires and transistors. A second, shorter delay after "You get . . ." heightens the suspense. Adjust these time loops so that they don't create too long a wait.

Prize Routine

```
10 REM EXAMPLE OF PRIZE ROUTINE
1000 PRINT:PRINT:PRINT "YOU MAY SELECT"
1010 PRINT:PRINT "A PRIZE BY"
1020 PRINT:PRINT "PRESSING ANY"
1030 PRINT:PRINT "LETTER FROM A TO Z."
1040 PRINT:PRINT:PRINT
1050 GET P$:REM SEE TEXT REGARDING THIS LINE
1060 IF P$="" THEN 1050
1070 PRINT "{CLR}":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
1080 PRINT:PRINT:PRINT "I AM SEARCHING."
1090 FOR T=1 TO 1200:NEXT T
1100 PRINT:PRINT "YOU GET. . .":PRINT:PRINT
1110 FOR T=1 TO 800:NEXT T
1120 IF P$="A" THEN PRINT "A HILL OF FIRE ANTS"
1130 IF P$="B" THEN PRINT "AN OVERRIPE BANANA"
1140 IF P$="C" THEN PRINT "SIX COCKROACHES"
1150 IF P$="D" THEN PRINT "DRY TERMITES"
1160 IF P$="E" THEN PRINT "A HUMID DAY"
1170 IF P$="F" THEN PRINT "SUMMER FLOODS"
1180 IF P$="G" THEN PRINT "7 BAGS OF GRASS CLIPPINGS"
1190 IF P$="H" THEN PRINT "WINTER GUESTS"
1200 IF P$="I" THEN PRINT "2 MOLE CRICKETS"
1210 IF P$="J" THEN PRINT "SEAS 3 TO 4 FEET"
1220 IF P$="K" THEN PRINT "STINGRAY TAILS"
1230 IF P$="L" THEN PRINT "LAWN BLIGHT"
1240 IF P$="M" THEN PRINT "A SWARM OF MOSQUITOES"
1250 IF P$="N" THEN PRINT "GREEN MILDEW"
1260 IF P$="O" THEN PRINT "TRAFFIC JAMS"
1270 IF P$="P" THEN PRINT "2 PALMETTO BUGS"
1280 IF P$="Q" THEN PRINT "A VIEW OF CONDOS"
1290 IF P$="R" THEN PRINT "RED ANTS"
1300 IF P$="S" THEN PRINT "SUNBURN"
1310 IF P$="T" THEN PRINT "AFTERNOON THUNDERSTORM"
1320 IF P$="U" THEN PRINT "SAND IN YOUR SHOES"
1330 IF P$="V" THEN PRINT "A BUCKET OF RED TIDE"
1340 IF P$="W" THEN PRINT "WINTER DROUGHT"
1350 IF P$="X" THEN PRINT "HIGHER TAXES!"
1360 IF P$="Y" THEN PRINT "YELLOW MOLD"
1370 IF P$="Z" THEN PRINT "100 NEMATODES"
1380 END
```

B. *Audiovisual Reward on VIC-20*

If you are working with a VIC-20 you can add the next audiovisual routine to the beginning of the prize routine above.

The music in this one will work without any special cartridge, provided you have the available memory. The music is contained in the DATA statements and is unique in that it plays a duet, two voices singing simultaneously. It is a waltz, and each line of DATA contains three beats, except for the first and the last. The first adds a beat of silence, and the last is a "stinger" followed by silence. The numbers are read in pairs (X,Y). You will find the music rather soothing when it comes as a reward for a game well played.

Lines 1030 and 1040 assume you will tack it into the prize routine above. If you don't, eliminate these two lines or reword them to suit your own needs.

The random placement of color is a stock VIC-20 routine, but it is especially effective in time with the music.

Audiovisual Reward (VIC-20)

```
10 REM  EXAMPLE OF AUDIOVISUAL REWARD ROUTINE ON VIC-20
1000 S2=36875:S3=36876:V=36878
1010 GOSUB 4000
1020 POKE V,4
1030 PRINT "■■■■■■■■  WHEN MUSIC ENDS"
1040 PRINT "■■■   PICK A TREAT!"
1050 FOR M=1 TO 48
1060 R=INT(RND(1)*500)+1
1070 READ X,Y
1080 C=INT(RND(1)*8)+1
1090 POKE S3,X:POKE S2,Y
1100 POKE 7680+R,160
1110 POKE 38400+R,C
1120 FOR B=1 TO 165:NEXT B
1130 NEXT M
1140 GOSUB 4000
1150 END
3000 DATA 0,0,147,219,147,219,147,219
3010 DATA 159,199,159,199,159,199
3020 DATA 147,201,163,201,183,201
3030 DATA 147,201,163,201,183,201
3040 DATA 175,228,175,228,175,228
3050 DATA 183,212,183,212,183,212
3060 DATA 147,215,167,215,201,215
3070 DATA 147,215,167,215,147,215
3080 DATA 143,219,143,219,143,219
3090 DATA 143,207,175,207,183,207
3100 DATA 147,209,147,207,147,201
3110 DATA 167,215,147,219,175,221
3120 DATA 183,209,147,209,163,209
3130 DATA 143,207,143,207,175,207
3140 DATA 147,201,163,201,183,201
3150 DATA 147,228,0,0
4000 FOR M=S2 TO V
4010 POKE M,0:NEXT
4020 RETURN
```

C. *Ego Reward*

Let's return now to a reward in universal BASIC. This one is calculated to inflate any player's ego. If you are able to add some color and audio to it, you will have a prize that will make players strive to win.

The subroutine in line 2000 is nothing more than the familiar time loop. Calling upon it many times builds up a feeling of breathless excitement. Once the winner's initials are entered, the computer begins a series of cheers that is almost exhausting.

The spaces in lines 1170, 1290, 1330, 1360, and 1560 are not critical. They merely position the various remarks in different spaces on the screen. You may be able to use TAB statements for the same purpose.

The illusion that the computer has tired itself with cheering the winner begins in line 1420. This one may deserve an Academy Award. The celebration stops. The screen is blank for a moment. Then the computer says, "Phew! (Pause) I'm tired . . ." This is followed by yet another pause. Then the computer, recognizing the worthiness of the recipient of the applause, agrees to one more shout. Lines 1530 and 1540 produce a spectacular double pause on a blank screen. This suspense is followed with an anticlimactic "Whoopee!" placed in loneliness on the screen.

Expand this idea all you can. Your players will love you for it.

Ego Reward

```
10 REM  "EGO REWARD ROUTINE"
1000 PRINT"□":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
1010 PRINT:PRINT:PRINT "CONGRATULATIONS!"
1020 GOSUB 2000
1030 PRINT:PRINT:PRINT "YOU DID IT!"
1040 GOSUB 2000
1050 PRINT:PRINT:PRINT "NOW DO ONE MORE THING."
1060 GOSUB 2000
1070 PRINT:PRINT:PRINT "GIVE ME YOUR INITIALS."
1080 PRINT:PRINT:PRINT
1090 GOSUB 2000
1100 INPUT "WHAT ARE THEY";N$
1105 PRINT "□":REM  REPLACE AS ABOVE.
1110 PRINT:PRINT N$;" IS TERRIFIC!"
1120 GOSUB 2000
1130 PRINT:PRINT "LET'S HEAR IT FOR ";N$;"!"
1140 GOSUB 2000
1150 PRINT:PRINT
1160 FOR H=1 TO 25
1170 PRINT "HOORAY      ";
1180 NEXT H
1190 PRINT
1200 PRINT:PRINT
1210 FOR H=1 TO 25
1220 PRINT "FOR ";N$;"!"
1230 NEXT H
1240 FOR H=1 TO 10
1250 PRINT:PRINT:PRINT N$;" KNOWS THE BIBLE!"
1260 NEXT H
1270 GOSUB 2000
1280 FOR H=1 TO 20
1290 PRINT "          ";N$
1300 NEXT H
1310 FOR H=1 TO 20
1320 PRINT N$
1330 PRINT "               ";N$
1340 NEXT H
1350 FOR H=1 TO 100
1360 PRINT "*   ";
1370 NEXT H
1380 FOR H=1 TO 10
1390 PRINT "YOU GAVE IT YOUR BEST,"
1400 PRINT N$;", YOU DID IT!"
1410 NEXT H
1415 GOSUB 2000
1420 PRINT "□":REM  REPLACE AS ABOVE.
1430 GOSUB 2000
1440 PRINT:PRINT:PRINT "PHEW!"
1450 GOSUB 2000
```

```
1460 PRINT:PRINT "I'M TIRED..."
1470 GOSUB 2000
1480 PRINT:PRINT "BUT ";N$;" DESERVES IT."
1490 GOSUB 2000
1500 PRINT:PRINT "JUST ONE MORE."
1510 GOSUB 2000
1520 PRINT "□":REM  REPLACE AS ABOVE.
1530 GOSUB 2000
1540 GOSUB 2000
1550 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
1560 PRINT "       WHOOPEE!"
1570 PRINT:PRINT:PRINT:PRINT:PRINT
1580 END
2000 FOR T=1 TO 1100:NEXT T
2010 RETURN
```

D. *Fruit of the Spirit Reward*

Giving a winning player an opportunity to pick one of the "Fruits of the Spirit" as listed in Galatians 5:22–23 will be both rewarding and educational. Right away, it will teach that there are nine virtues mentioned in Paul's letter. Knowing them in order is not one of the most important things for Christians to add to their skills, but it can't hurt! A time-delay loop in line 6010 holds the identifying statement on the screen for an instant, giving the player a moment to anticipate the answer. Eventually, the player will know whether it will be "love" or "humility" before it is ever displayed. Line 6020 causes the answer to be printed fifty times before it quits. You can change it to a greater or lesser number to suit yourself.

The numbers in this routine are intentionally high. If you must raise them still higher, be very careful to correct the line numbers in line 5070. This line is extremely important if you want to see the correct answers.

With a little effort, you can expand this program into a fully developed independent game. Try, for instance, using it to teach the Ten Commandments or the Twelve Tribes of Israel. Adding more rounds and scoring should be no difficulty for you if you have absorbed the lessons in the earlier programs.

Fruit of the Spirit Reward

```
10 REM "FRUIT OF THE SPIRIT" REWARD ROUTINE
5000 PRINT:PRINT "CONGRATULATIONS!  FOR"
5010 PRINT:PRINT "A PRIZE, YOU MAY PICK"
5020 PRINT:PRINT "A FRUIT OF THE SPIRIT."
5030 PRINT:PRINT "(GAL. 5:22-23)"
5040 PRINT:PRINT:PRINT "WHICH WILL YOU PICK":PRINT
5050 INPUT "FROM 1 TO 9";F
5060 IF F>9 OR F<1 THEN 5050
5070 ON F GOTO 5200, 5250, 5300, 5350, 5400, 5450, 5500, 5550, 5600
5200 A$="FIRST"
5210 B$="*****   LOVE   *****"
5220 GOSUB 6000
5250 A$="SECOND"
5260 B$="*****   JOY   *****"
5270 GOSUB 6000
5300 A$="THIRD"
5310 B$="*****   PEACE   *****"
5320 GOSUB 6000
5350 A$="FOURTH"
5360 B$="****  PATIENCE  ****"
5370 GOSUB 6000
5400 A$="FIFTH"
5410 B$="****  KINDNESS  ****"
5420 GOSUB 6000
5450 A$="SIXTH"
5460 B$="****  GOODNESS  ****"
5470 GOSUB 6000
5500 A$="SEVENTH"
5510 B$="*** FAITHFULNESS ***"
5520 GOSUB 6000
5550 A$="EIGHTH"
5560 B$="***  HUMILITY  ***"
5570 GOSUB 6000
5600 A$="NINTH"
5610 B$="*** SELF-CONTROL ***"
5620 GOSUB 6000
6000 PRINT:PRINT "THE ";A$;" FRUIT IS:"
6010 FOR T=1 TO 1800:NEXT T
6020 FOR P=1 TO 50
6030 PRINT B$
6040 NEXT P
6050 END
```

E. *Psychic Computer*

"The Psychic Computer" is a quick little reward that never disappoints the player. The important thing to point out in your instructions is that the calculation is *not* to be done on the computer. The whole point is that the computer is kept in the dark. The seven steps toward an answer should be done on paper. Only the final answer is to be entered into the computer at line 250.

The formulas in lines 260 and 270 break the player's "bottom line" into two parts. The first number (1 to 12) becomes the variable *X*, which line 280 kicks down to the correct month. The second number is always the player's admitted age and becomes the variable *Y*. When line 470 prints the results on the screen, there is usually a little astonished gasp.

We can thank my son, David, for working out the math in lines 260 and 270 that makes the magic. The math in the paper calculation is an ancient parlor game.

Psychic Computer

```
10 REM  "PSYCHIC COMPUTER" REWARD ROUTINE
100 PRINT:PRINT "WHAT IS THE NUMBER OF YOUR MONTH OF BIRTH (1 TO 12)?"
110 GOSUB 1000
120 PRINT:PRINT "MULTIPLY BY 2"
130 GOSUB 1000
140 PRINT:PRINT "ADD 5"
150 GOSUB 1000
160 PRINT:PRINT "MULTIPLY BY 50"
170 GOSUB 1000
180 PRINT:PRINT "ADD YOUR AGE"
190 GOSUB 1000
200 PRINT:PRINT "SUBTRACT 365"
210 GOSUB 1000
220 PRINT:PRINT "ADD 115"
230 GOSUB 1000
240 PRINT:PRINT
250 INPUT "ENTER RESULT";N
260 X=INT(N*.01)
270 Y=INT((N*.01-X)*100+.5)
280 ON X GOTO 300, 310, 320, 330, 340, 350, 360, 370, 380, 390, 400, 410
300 A$="JANUARY":GOTO 450
310 A$="FEBRUARY":GOTO 450
320 A$="MARCH":GOTO 450
330 A$="APRIL":GOTO 450
340 A$="MAY":GOTO 450
350 A$="JUNE":GOTO 450
360 A$="JULY":GOTO 450
370 A$="AUGUST":GOTO 450
380 A$="SEPTEMBER":GOTO 450
390 A$="OCTOBER":GOTO 450
400 A$="NOVEMBER":GOTO 450
410 A$="DECEMBER":GOTO 450
450 PRINT "⌂":REM REPLACE THIS LINE WITH YOUR CLEAR SCREEN COMMAND IF NECESSARY
460 PRINT:PRINT:PRINT:PRINT
470 PRINT "YOU WERE BORN IN ";A$;",";Y;"YEARS AGO."
480 END
1000 FOR T=1 TO 2000: NEXT T
1010 RETURN
```

F. *I've Got Your Number!*

While we are remembering parlor games, here is a computerized version of a famous one. "I've Got Your Number!" again asks that all the calculation be done outside the computer. The point is that the answer be kept a secret from the machine. The steps are simple enough for most players to do the calculations in their heads.

Line 120 is important. It prevents a player from following the natural tendency to type in the answers. All the INPUT is waiting for is an indication that the step has been completed. A simple pressing of the return/enter key will tell the player what to do next.

Since the answer is always 3, no internal calculation is necessary. Just put it in a PRINT statement like the one in line 250.

In sixteen lines you can give the winning player a little harmless fun.

I've Got Your Number!

```
10 REM  "I'VE GOT YOUR NUMBER" REWARD ROUTINE
100 PRINT:PRINT "I'VE GOT YOUR NUMBER!":PRINT "--------------------"
110 PRINT:PRINT "PICK A NUMBER, BUT"
120 PRINT "DON'T TELL ME."
130 INPUT "GOT IT";N
140 PRINT:PRINT "DOUBLE IT."
150 INPUT "READY";N
160 PRINT:PRINT "ADD 9"
170 INPUT "OK";N
180 PRINT:PRINT "SUBTRACT 3"
190 INPUT "GO ON";N
200 PRINT:PRINT "DIVIDE BY 2"
210 INPUT "NEXT";N
220 PRINT:PRINT "SUBTRACT YOUR NUMBER"
230 PRINT:PRINT "I'VE GOT THE ANSWER."
240 INPUT "DO YOU";N
250 PRINT:PRINT "THE ANSWER IS 3"
260 END
```

COMPUTERS/RELIGION

This breakthrough personal computer book applies the latest technology to open up the fascinating world of the Bible through appealing and exciting computer games.

Bible Basic

Bible Basic **serves up an irresistible menu of games for the home computer designed to appeal to players of all ages. Each game has been written to allow a beginning user immediate success, while the more experienced can modify the basic programs to put their skills to the test. The *Bible Basic* games are complete, ready-to-play, debugged models. They use BASIC computer language, and are compatible with most popular personal computer systems on the market, including the VIC-20™, Commodore 64™, TRS-80®, Apple II™, and Atari 400/800™. The programs avoid those areas of incompatibility between the BASIC languages employed by individual manufacturers.**

All *Bible Basic* games are designed for easy, accurate entry into personal computers for playing alone or with others—at home, at church, or in group meetings. Bernard Bangley provides programming details and sample programming runs for each game, as well as helpful tips on how users can modify the programs to create games of their own.

Bible Basic **is an invitation to plug into such games as "Jacob's Ladder," "Which Book?" "Password: Shibboleth," "David and Goliath," "Famous Bible Women," "Captive!" "The Perils of Paul," and many more. From the very first program, you'll find yourself discovering an exciting and entertaining way to learn about the Bible, while opening up incredible new worlds of computer gaming fun.**

And it's just as near as your personal computer.

Bernard K. Bangley **is minister of Pine Shores Presbyterian Church in Sarasota, Florida, and the author of several popular books, including *Growing in His Image.* The engineering of the programs in *Bible Basic* was fine-tuned by the author's seventeen-year-old son, David, who is his computer expert in residence.**

HARPER & ROW, PUBLISHERS