

FIRST COMPUTER LIBRARY



# SIMPLE BASIC

```
10 CLS  
20 FOR K=1 TO  
30 PRINT "HELLO"  
40 NEXT K
```



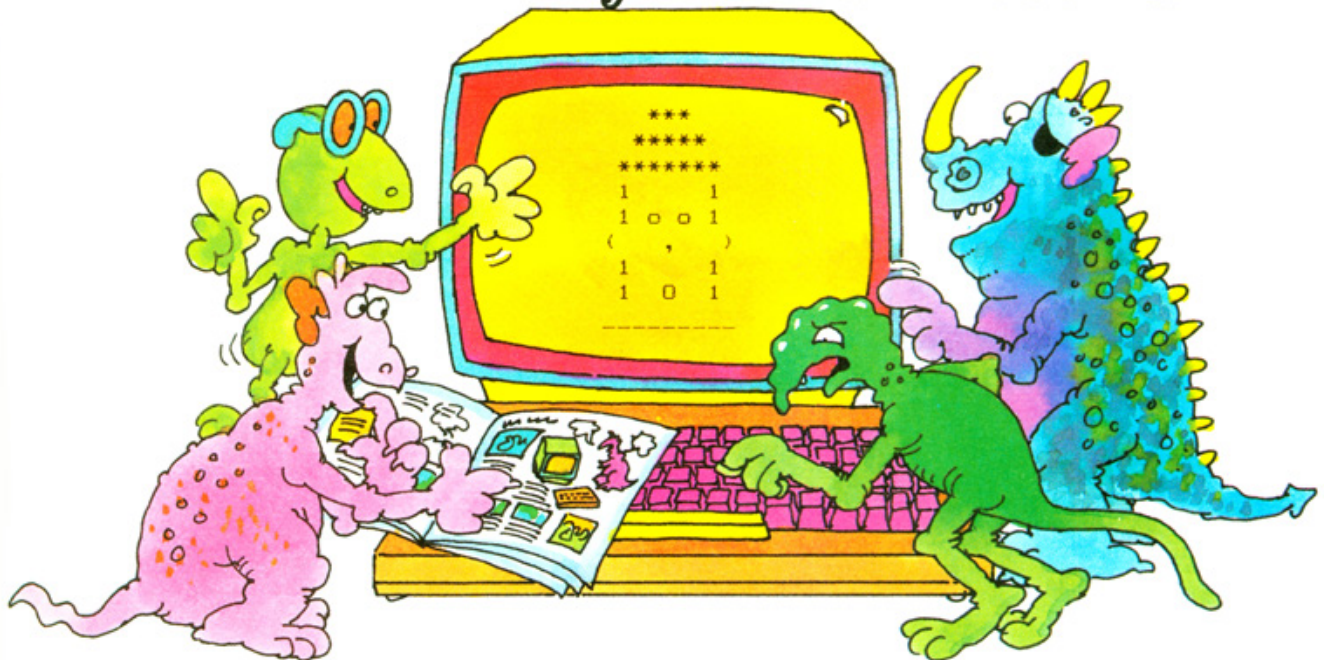
USBORNE

FIRST COMPUTER LIBRARY

# SIMPLE BASIC

Gaby Waters

Illustrated by Graham Round

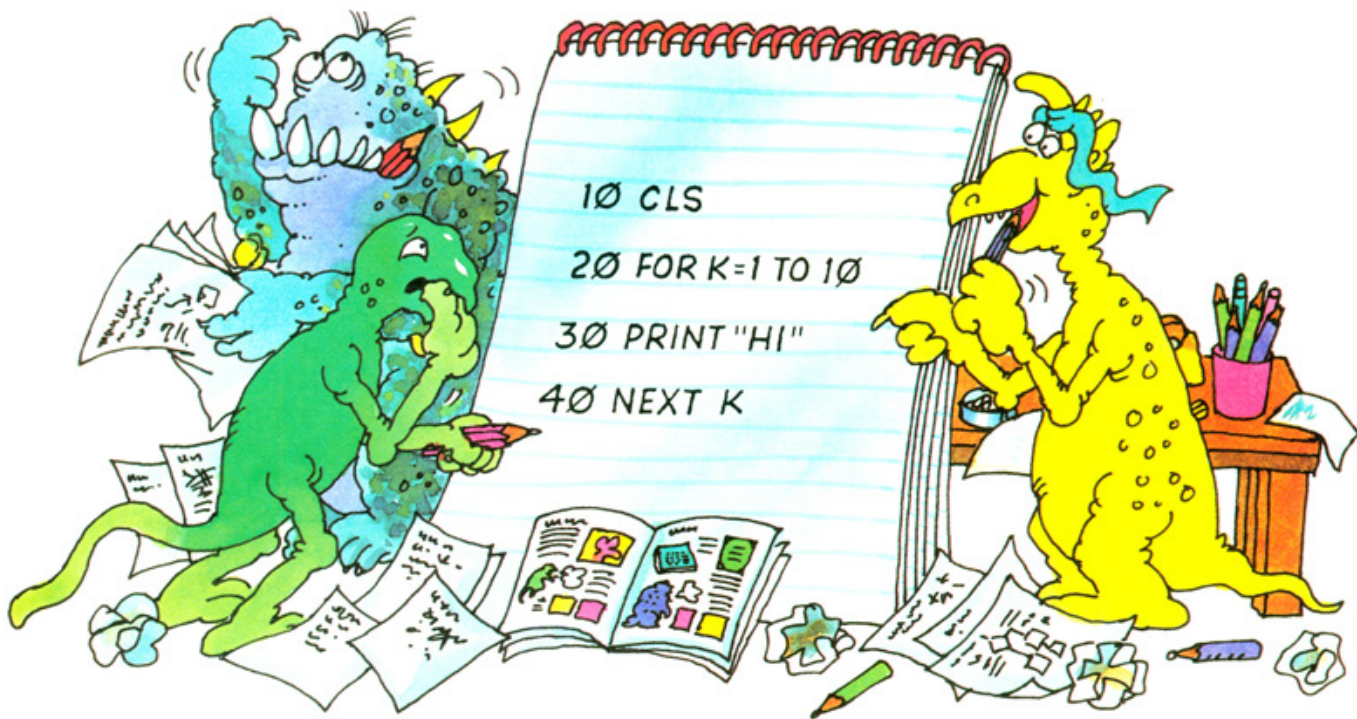


Additional programs by Paul Shreeve  
Series editor: Helen Davies



# Contents

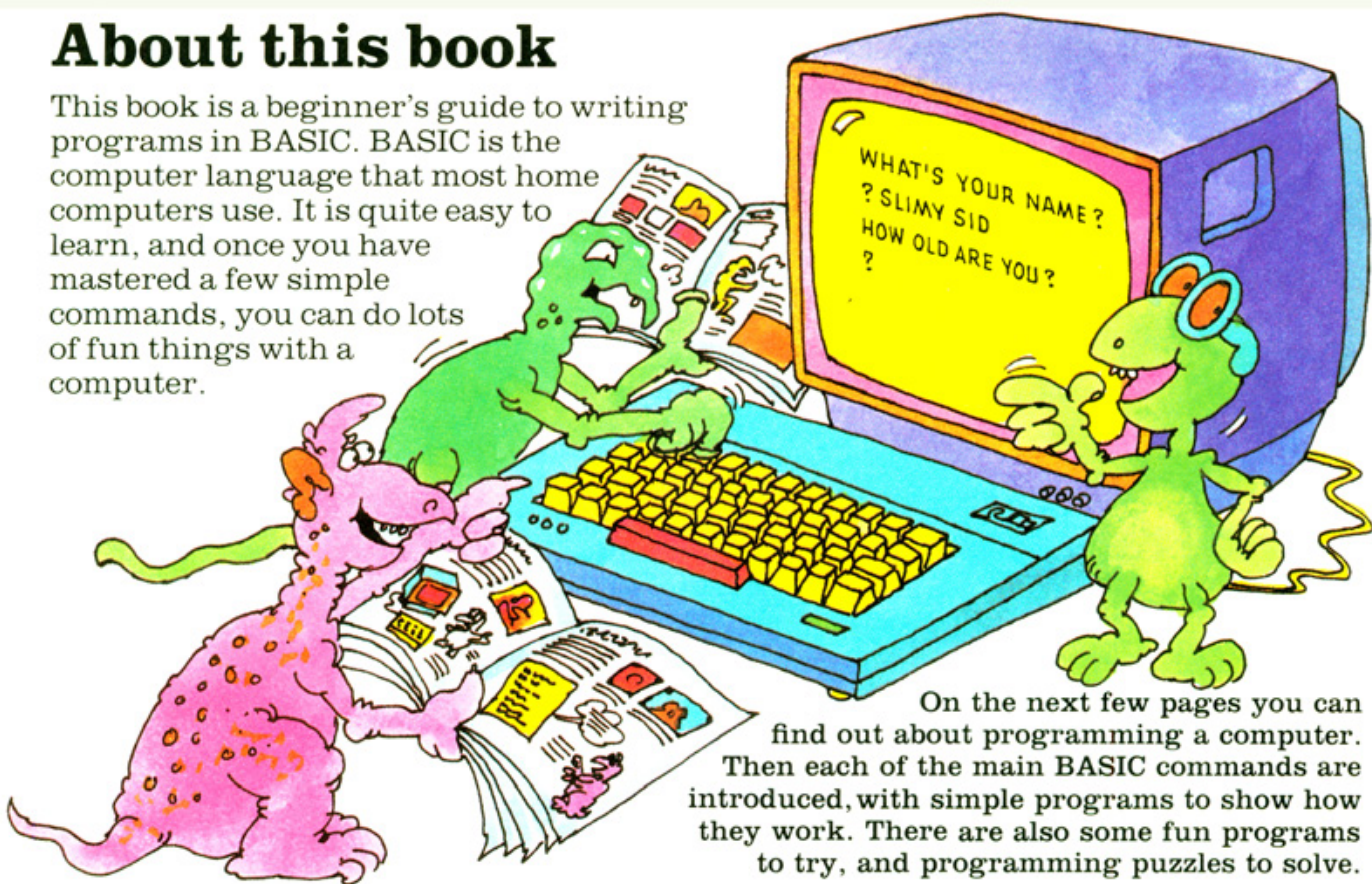
3	About this book	22	Comparing things
4	What is BASIC?	24	IF and THEN programs
6	First commands in BASIC	26	Making loops
8	Writing programs	28	Loopy programs
10	Giving the computer a program	30	Random numbers
12	Programs with PRINT	32	Debugging programs
14	Introducing variables	33	Command chart
16	More about variables	34	Programs and puzzles
18	Using INPUT	44	Puzzle answers
20	Fun with PRINT	48	Index



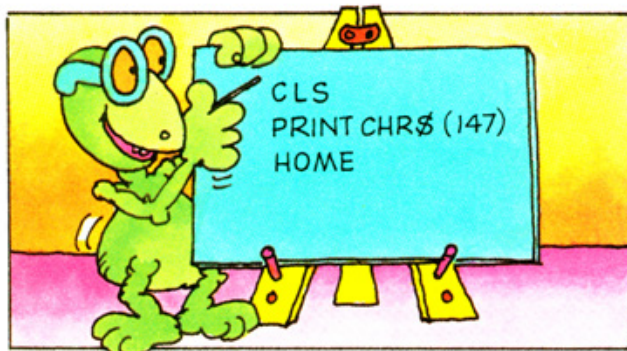


# About this book

This book is a beginner's guide to writing programs in BASIC. BASIC is the computer language that most home computers use. It is quite easy to learn, and once you have mastered a few simple commands, you can do lots of fun things with a computer.



On the next few pages you can find out about programming a computer. Then each of the main BASIC commands are introduced, with simple programs to show how they work. There are also some fun programs to try, and programming puzzles to solve.



One or two BASIC commands vary on different makes of computer. These are clearly marked where they occur. On page 33 there is a chart showing which version of these commands to use for your computer.

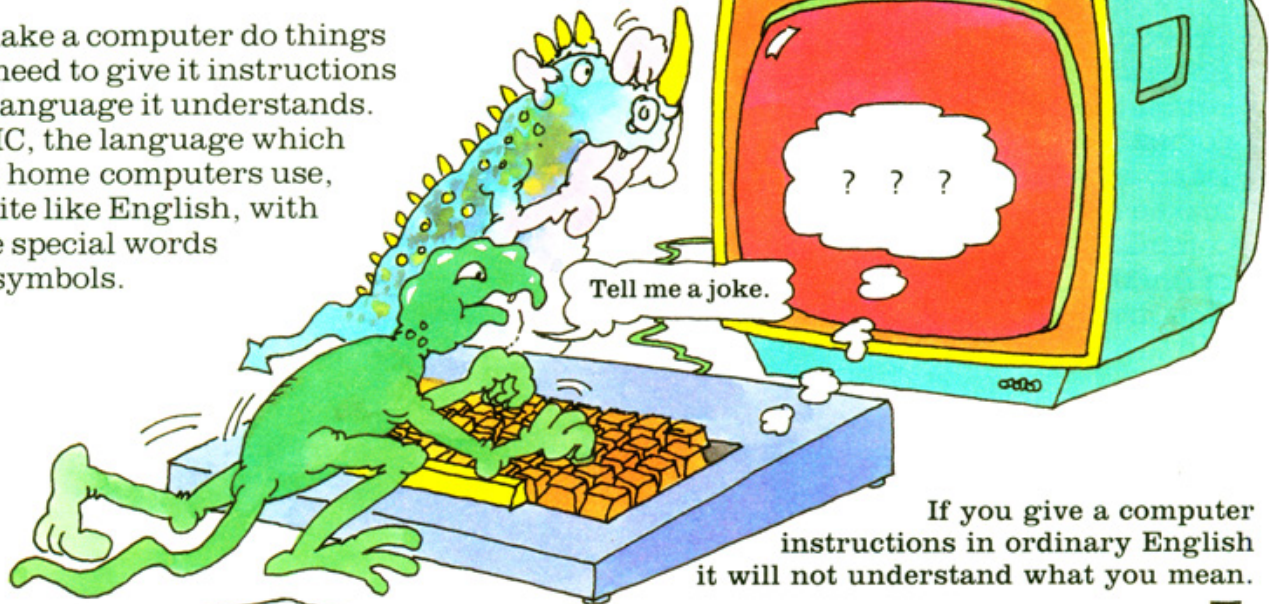


At the back of the book there are the answers to the puzzles, with simple explanations. There are also some tips and hints to help you if your programs do not work.

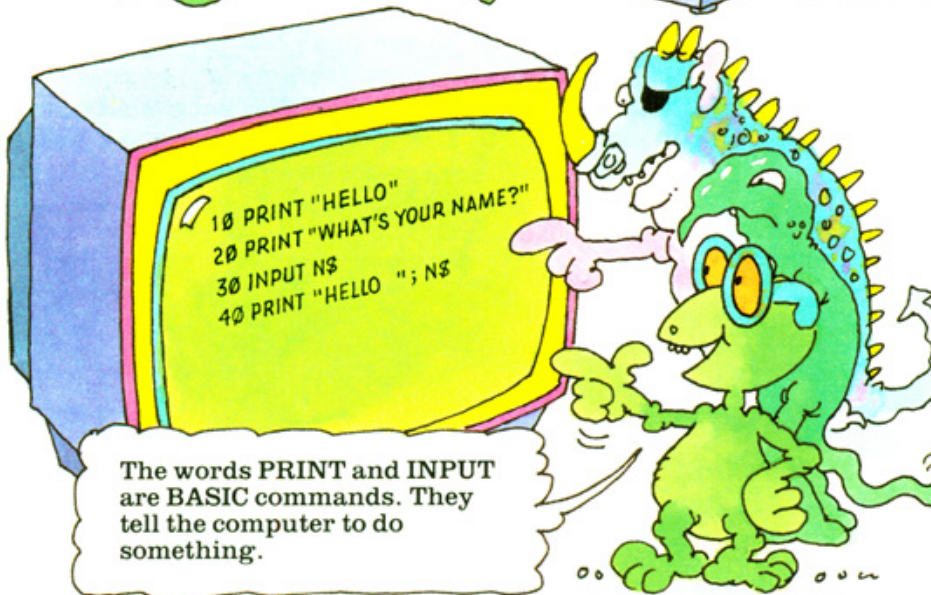


# What is BASIC?

To make a computer do things you need to give it instructions in a language it understands. BASIC, the language which most home computers use, is quite like English, with some special words and symbols.



If you give a computer instructions in ordinary English it will not understand what you mean.



The words PRINT and INPUT are BASIC commands. They tell the computer to do something.

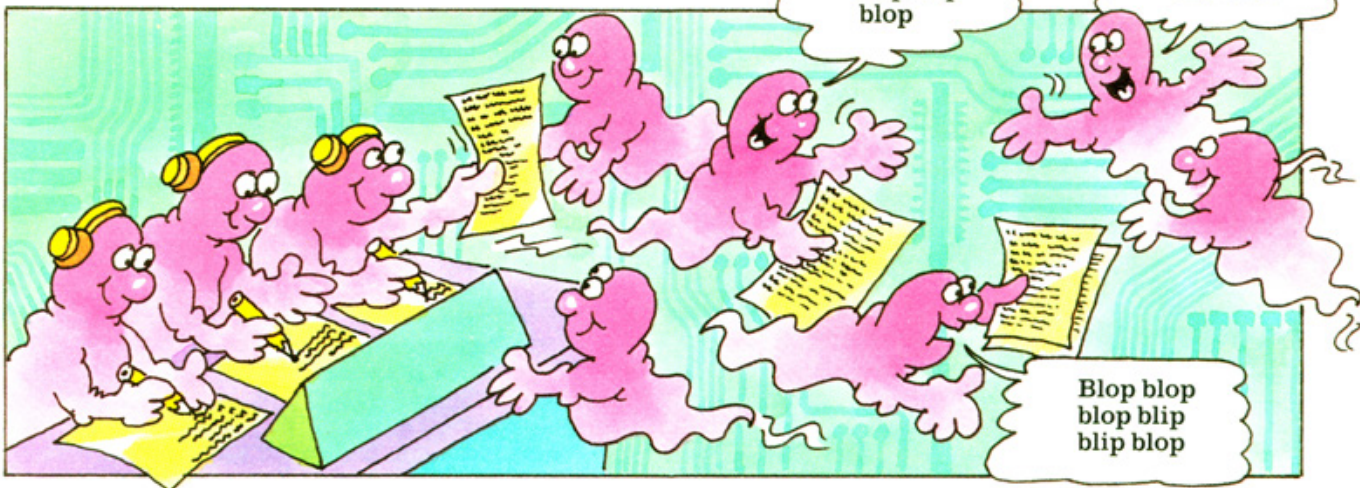
Giving a computer instructions is called programming and a list of instructions is called a program. On the screen above you can see what a program in BASIC looks like.



BASIC can be used to make a computer do all sorts of things, for instance, draw pictures, work out problems and even make funny noises.



## How computers work things out

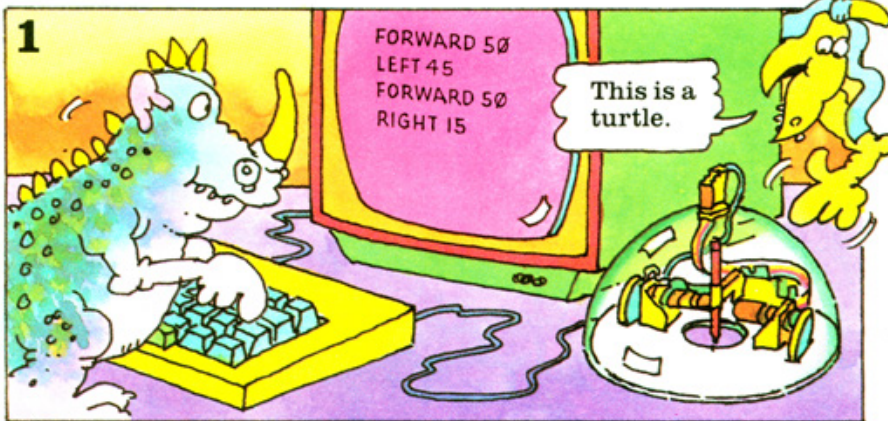


When you give a computer a program, it then has to translate the computer language into computer

code. Computer code is made up of electrical signals. Computers use this code to

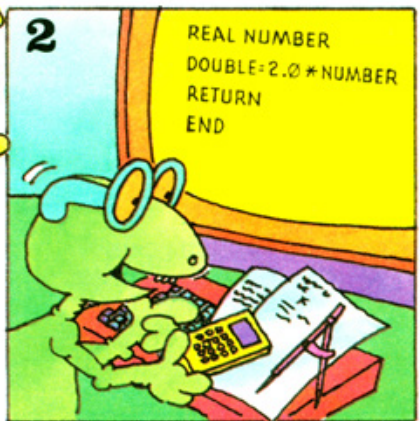
do all their work. You can write programs in computer code, but it is very difficult.

## Other computer languages



There are lots of other computer languages besides BASIC. The picture above shows the language LOGO which uses simple commands, such as LEFT

and RIGHT, to make a computer draw shapes on the screen. LOGO is also used to control a small robot called a turtle. It has a pen which draws shapes as it moves.



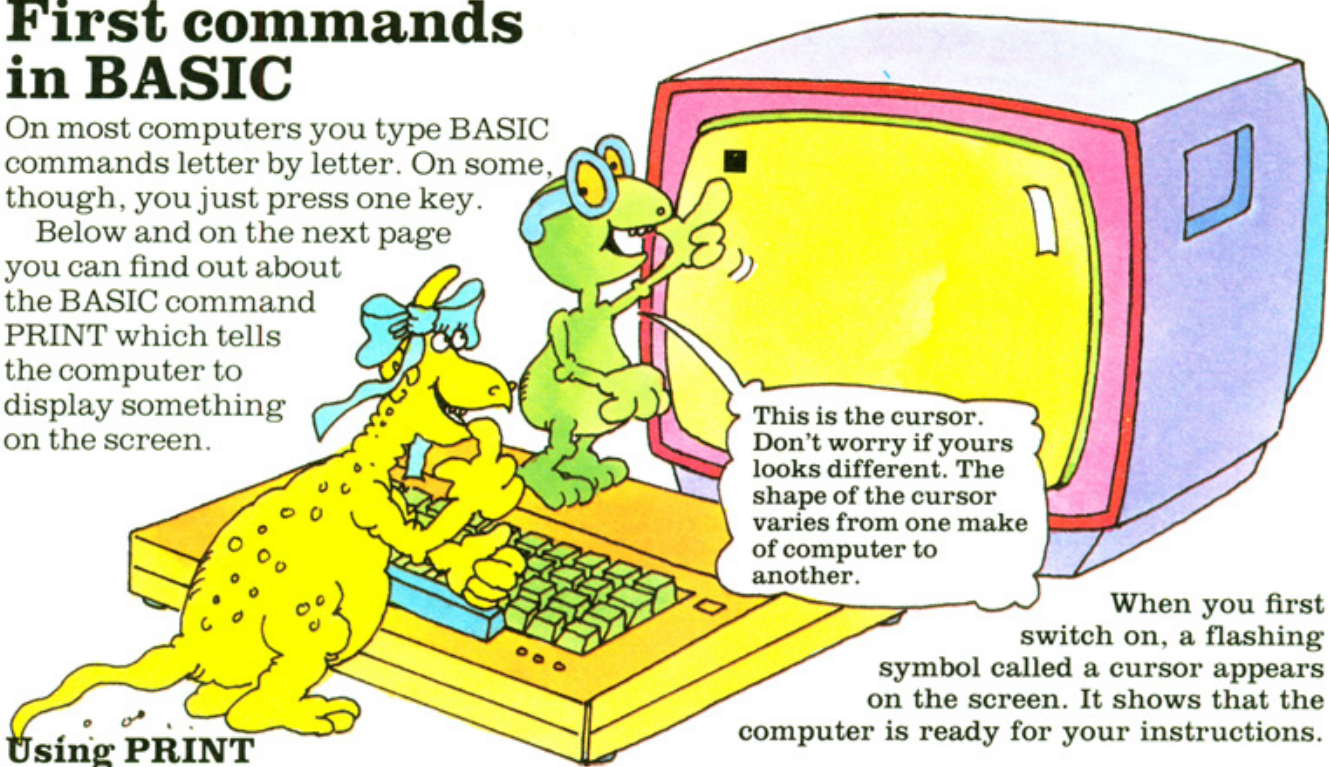
The instructions in this picture are written in a programming language called FORTRAN. FORTRAN is very good for working out maths and science problems.



# First commands in BASIC

On most computers you type BASIC commands letter by letter. On some, though, you just press one key.

Below and on the next page you can find out about the BASIC command PRINT which tells the computer to display something on the screen.



When you first switch on, a flashing symbol called a cursor appears on the screen. It shows that the computer is ready for your instructions.

## Using PRINT



To make a computer display a word on the screen, you type PRINT, followed by the word inside quotes. Type carefully, as mistakes will confuse the computer.



To make the computer carry out the command, press the key marked RETURN (or ENTER on some computers).



When you press RETURN (or ENTER), the computer puts the word on the screen. \* Try using PRINT to make your computer display words.



## Displaying messages

Below you can find out how to clear the screen after each message.

COMPUTERS  
ARE COOL

Slimy Sid  
is King

UGLY MONSTERS  
RULE OK

PRINT "MONSTERS FOREVER"  
MONSTERS FOREVER

Here are some ideas for messages to display on the screen. Use a PRINT command for each message and type the words inside quotes.

## Clearing the screen

**1** CLS  
HOME  
PRINT CHR\$(147)



These are some common clear screen commands.

**2** ■

The command to clear the screen varies on different makes of computer. You can find out which one your computer uses on page 33.\*

When you press RETURN after this command, the computer wipes everything from the screen except the cursor.

## Using PRINT with numbers

PRINT 1234

PRINT 999

PRINT 12

PRINT also makes the computer display numbers. You do not need to put numbers in quotes. Try some PRINT commands, like those above, to make the computer display numbers.

\*Wherever you see CLS in this book, use your computer's clear screen command.



# Writing programs

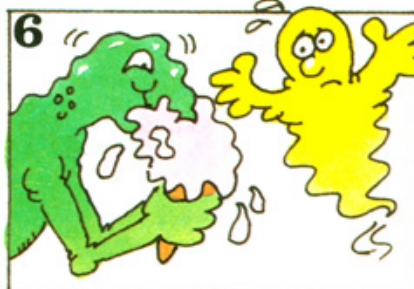
Imagine trying to make a computer-ghost buy an icecream. An instruction like the one in the picture would be too difficult for the ghost's computer brain. You need to give it a list of commands called a program.

Before you write the program, work out what the computer needs to do and write a plan in English.



## PLAN

1. Go to ice-cream stall
2. Ask for one large strawberry ice-cream
3. Pay ice-cream man
4. Take ice-cream (do not eat it)
5. Turn round and come back
6. Give ice-cream to Slimy Sid



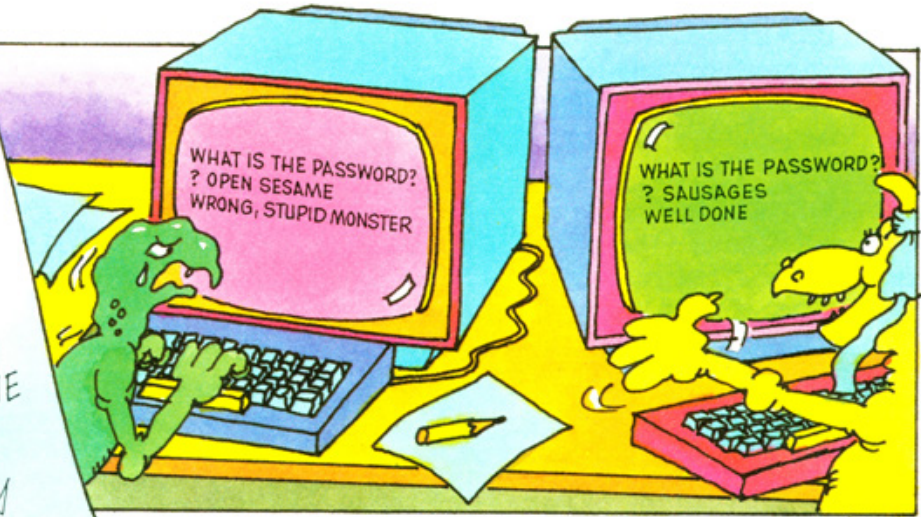
You have to work out exactly what the computer-ghost needs to do at every stage, and make a list of step-by-step instructions.



## Planning a program

### PROGRAM PLAN

1. Clear screen
2. Ask for password
3. Wait for answer
4. If answer is correct, display message WELL DONE
5. If answer is wrong, display rude message



When you write a computer program, first outline exactly what you want the computer to do. On the left is a plan for a password program.

← This command is different on some computers. See page 33.

```
10 CLS
20 PRINT "WHAT IS THE PASSWORD?"
30 INPUT P$
40 IF P$="SAUSAGES" THEN PRINT "WELL DONE"
50 IF P$<>"SAUSAGES" THEN PRINT "WRONG, STUPID MONSTER"
```

You can find out what the BASIC commands in this program mean, later in the book.

The line numbers usually go up in jumps of ten so you can fit in extra commands later.

To turn the plan into a program you translate each instruction into BASIC. You put each instruction on a separate line, starting with a line number. The numbers tell the computer the order of the instructions.





# Giving the computer a program

When you give a computer a program, it stores all the commands in its memory and does not carry them out until you tell it to.

On the right is a program to make the computer display a silly poem. Type it into your computer using the hints below to help you.

On most computers the number 0 has a line through it, so you don't confuse it with the letter O.

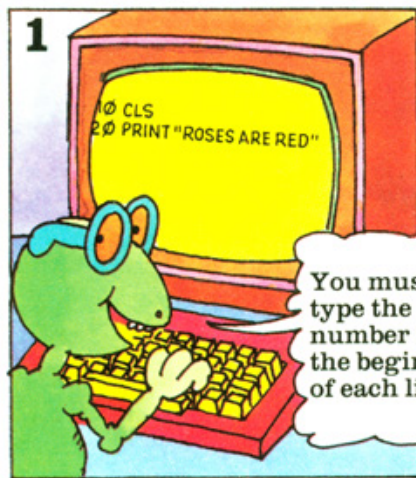
## Poem program

Remember that this command is different on some computers.

```
10 CLS  
20 PRINT "ROSES ARE RED"  
30 PRINT "VIOLETS ARE BLUE"  
40 PRINT "WITH A FACE LIKE YOURS"  
50 PRINT "YOU BELONG IN A ZOO"
```

The program uses PRINT commands to make the computer display the words of the poem.

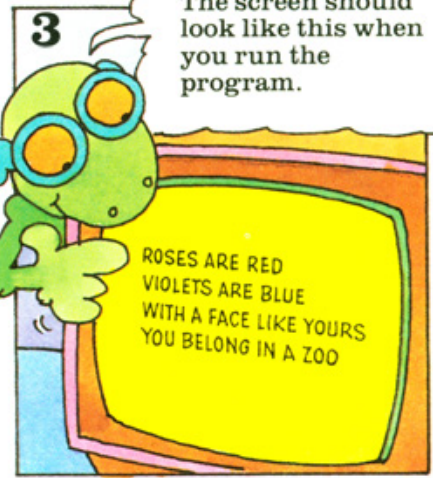
## Typing in a program



You must type the line number at the beginning of each line.



Your computer's DELETE key may have a different name. Check page 33, or your manual.



The screen should look like this when you run the program.

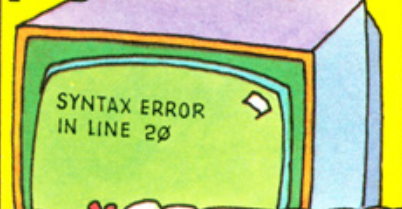
Type the program line by line. At the end of each line, check there are no mistakes, then press RETURN. This makes the computer store the line in its memory.

If you make a typing mistake, you can rub it out by pressing the DELETE key (sometimes labelled DEL for short).

To make the computer carry out the program, you type the command RUN and press RETURN. This is called running the program.



## Mistakes in programs



The computer may display an error message like this.

If a program does not run properly, you have probably made a mistake in it. Even a typing error can stop a program running. Mistakes in programs are called bugs. You can find out how to correct them on page 32.

## More about programs



You can type program lines in any order you like. Try typing in this program and running it. The computer will sort out the lines and carry them out in order.

```
Birthday party program
30 PRINT "PLEASE COME TO MY"
70 PRINT "NICE PRESENT!"
60 PRINT "AND BRING ME A"
20 PRINT "DEAR UGLYMUG"
50 PRINT "ON SATURDAY"
10 CLS
80 PRINT "LOVE SLIMY SID"
40 PRINT "BIRTHDAY PARTY"
```

You can also make the computer display or "list" the program lines in the right order. To do this you type LIST and press RETURN.

## Getting rid of programs



Some computers clear the screen when you type NEW.

To get rid of a program you type the command NEW and press RETURN. This wipes the program from the computer's memory.

To change a line you just type a new version with the same line number.

20

70 PRINT "A CHOCOLATE CAKE"

75 PRINT "(I LIKE CHOCOLATE)"

To get rid of a program line you type just its line number and press RETURN.

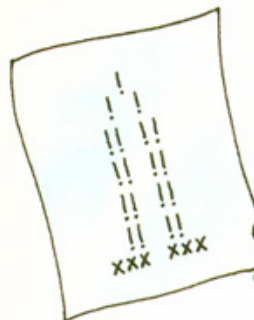
You can add an extra line anywhere in the program. Its line number tells the computer where it goes.

It is very easy to change computer programs by adding, altering or getting rid of program lines. Try making the changes shown above to the birthday party program.



# Programs with PRINT

You can use PRINT to display shapes and pictures made of symbols. The program on the right makes a rocket shape using symbols, letters and spaces. Try it on your computer.



## Rocket program

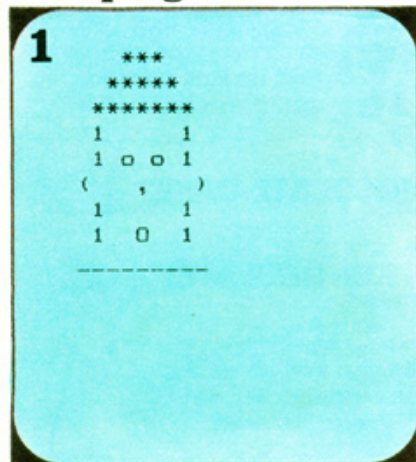
```
10 CLS
20 PRINT "ooo!"
30 PRINT "oo!o!"
40 PRINT "o!!o!!"
50 PRINT "o!!o!!"
60 PRINT "o!!o!!"
70 PRINT "o!!o!!"
80 PRINT "XXXoXXX"
```

This symbol means type a space. For each `o`, you should press the space bar or key, once.

Each PRINT command displays a line of symbols and spaces. The symbols and spaces must be inside quotes.

Remember to use your computer's clear screen command.

## Face program

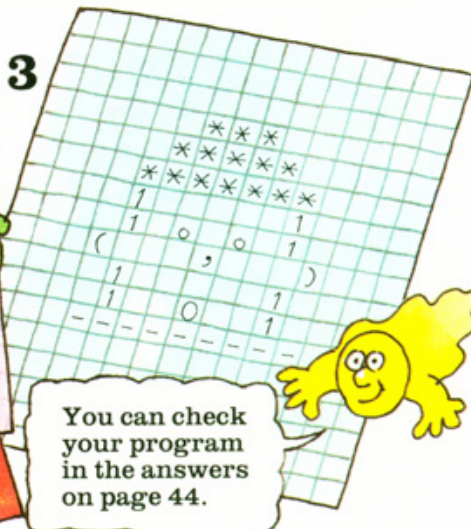


2



Put a clear screen command at the beginning of the program.

3



You can check your program in the answers on page 44.

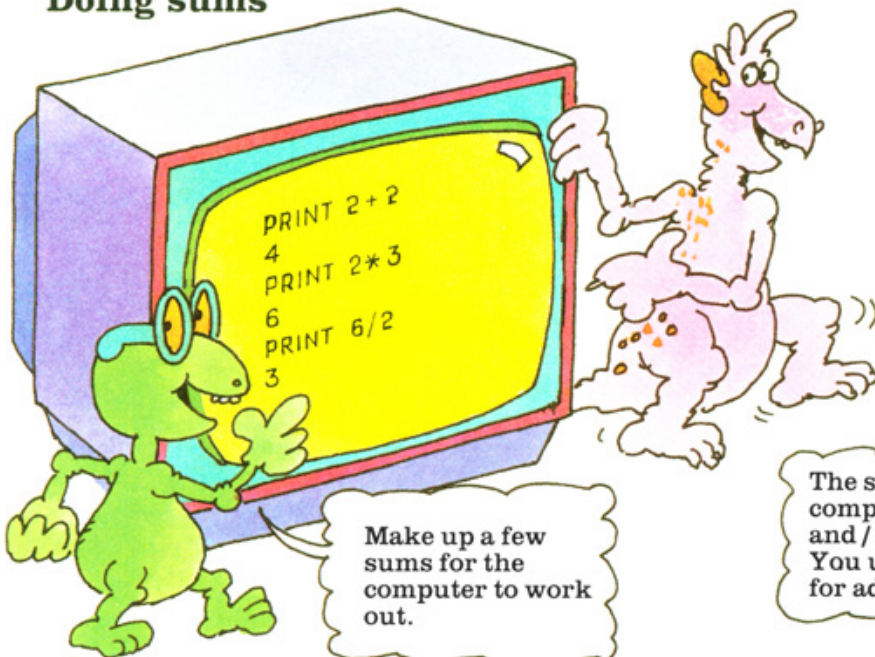
See if you can write a program to make your computer display a face like the one above.

In the program you will need a PRINT command for each line of the picture. It may help to draw the face on

squared paper first. Then you can work out how many symbols and spaces you need in each line.



## Doing sums



```
PRINT 2 + 2
4
PRINT 2 * 3
6
PRINT 6 / 2
3
```

Make up a few sums for the computer to work out.

The command **PRINT**, followed by a sum, tells the computer to work out the sum and put the answer on the screen. Try the **PRINT** commands shown on the left. The commands have no line numbers, so the computer carries out each one straight away.

The symbol, **\***, tells the computer to multiply, and **/** tells it to divide. You use the normal symbols for adding and subtracting.

## Hours program

```
10 CLS
20 PRINT "THERE ARE"
30 PRINT 365 * 24
40 PRINT "HOURS IN"
50 PRINT "A YEAR"
```

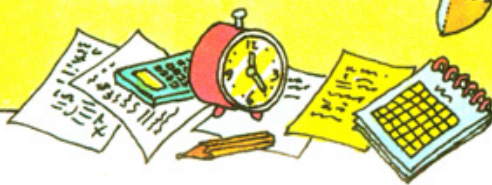
24 is the number of hours in a day.

365 is the number of days in a year.

THERE ARE  
8760  
HOURS IN  
A YEAR

This is what your screen will look like when you run the program.

This program makes the computer work out the number of hours in a year. Line 30 does the calculation. It multiplies the number of days in a year, by the number of hours in a day. Try running the program on your computer.

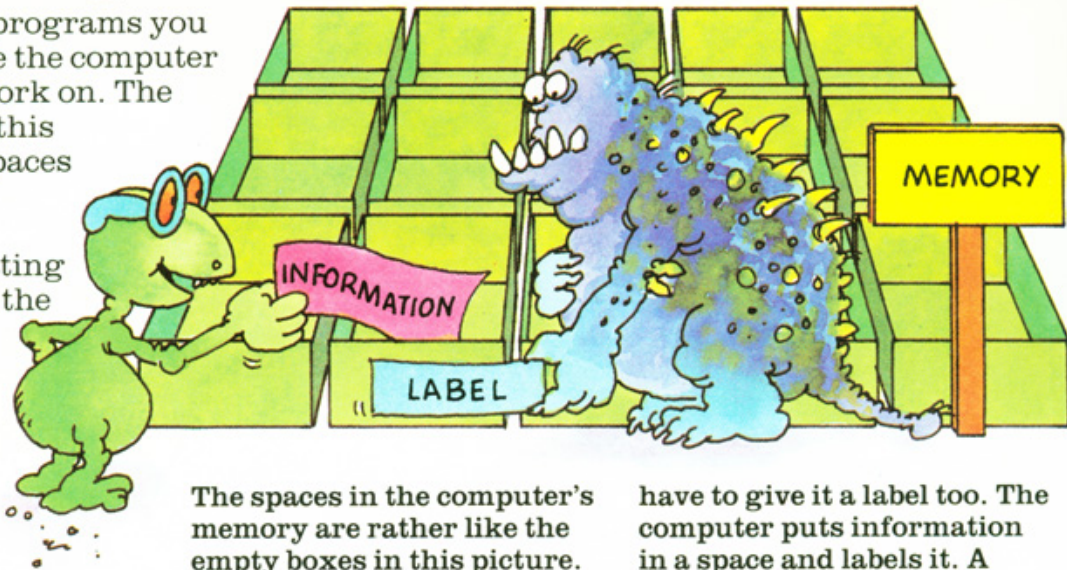




# Introducing variables

When you write programs you often need to give the computer information to work on. The computer stores this information in spaces called variables in its memory.

One way of putting information into the computer's memory is with the command **LET**.



The spaces in the computer's memory are rather like the empty boxes in this picture. When you give the computer a piece of information you

have to give it a label too. The computer puts information in a space and labels it. A labelled space is called a variable.

## Using LET



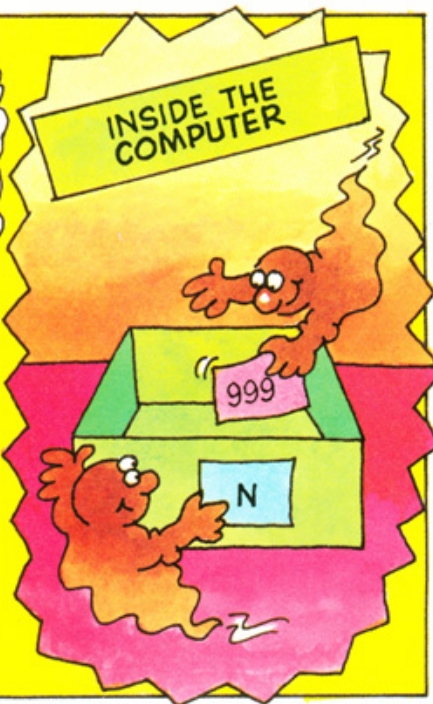
This is the label for the variable.

**LET N=999**

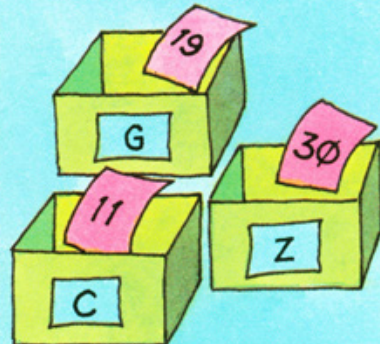
This is the number to be stored in the variable.



The **LET** command above tells the computer to store the number 999 in a variable labelled **N**.



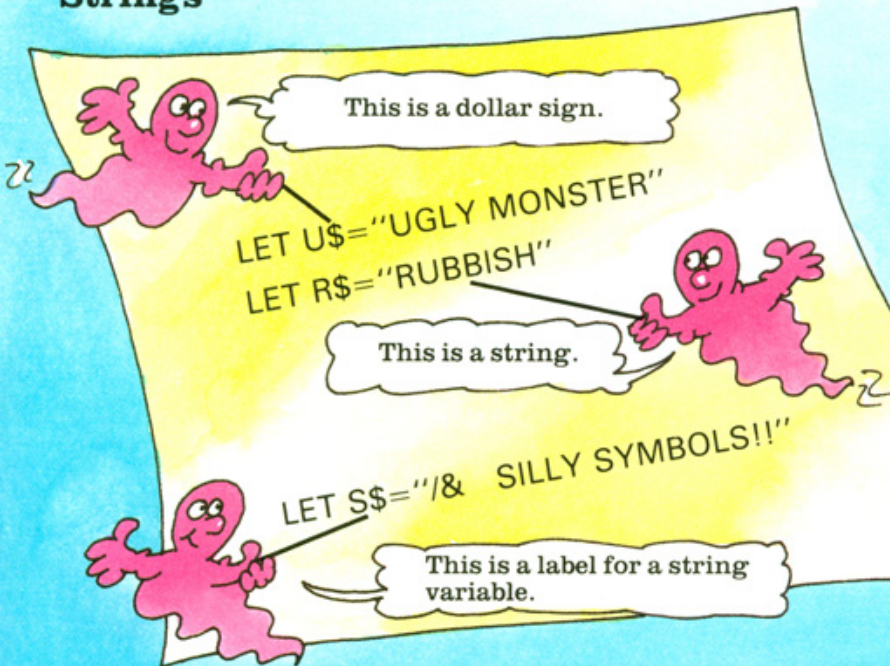
## LET puzzle



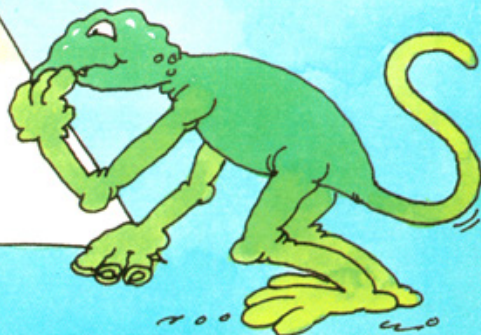
On a piece of paper, try writing three **LET** commands to store the numbers in the variables shown above. If you get stuck, look at the answers on page 44.



## Strings



In BASIC, words, letters and symbols are called strings. When you store a string in a variable, you must type the string in quotes. A variable that contains a string is called a string variable. Its label always has a dollar sign at the end.



### Using PRINT with variables

```
10 CLS
20 LET N=99
30 LET B$="BANANAS"
40 PRINT N
50 PRINT B$
```



The PRINT commands tell the computer to display the contents of the variables N and B\$.

To make the computer display the information stored in a variable, you use PRINT followed by the variable name. Try running the program above to see how this works.

### PRINT puzzle

```
10 CLS
20 LET A$="BRAINBOX"
30 LET B$="IS"
40 LET C$="BORING"
```

See if you can add three PRINT commands to this program to make the computer display the information stored in A\$, B\$ and C\$. The answer is on page 44.



# More about variables

Below and on the next page you can find out more about naming variables and how to use them in programs.

## Variable names

```
LET N1=24  
LET N2=48  
LET $$="STRAWBERRIES"  
LET NAMES$="BRAINBOX"
```

It's a good idea to give a variable a name that will help you remember what it contains.



Letters of the alphabet are usually used as names for variables. On most computers though, you can also use a letter with a

number, or even a short word. Your computer manual tells you what names your computer allows.

## Words and variables together

1

```
10 CLS  
20 LET N=7  
30 PRINT N;" SLIMY SLUGS"  
40 PRINT "ATE ";N;" SOGGY SANDWICHES"
```

These spaces make the computer leave a space on the screen between the words and the contents of the variable.

This is a semicolon.

You often need to display words next to the contents of a variable. To do this you use PRINT followed by the

variable name and the words, separated by a semicolon. In BASIC a semicolon tells the

## Illegal names



You must be careful not to use names that contain BASIC commands like these. They will confuse the computer and you will get a bug in your program.

2

```
7 SLIMY SLUGS  
ATE 7 SOGGY SANDWICHES
```

This is what your screen will look like when you run the program.

computer to display the next piece of information on the same line.



## Sums with variables

1

```
10 CLS
20 LET A=2
30 LET B=5
40 PRINT A+B
50 PRINT A*B
```

These are the answers to the sums.



2

```
35 PRINT A;"+";B;"=";
45 PRINT A;"x";B;"=";
```

2+5 = 7  
2x5 = 10

You can also use a semi-colon at the end of a line to tell the computer not to start a new line.



You can make the computer do sums with numbers stored in variables. Try the program above. It makes the computer add and multiply the numbers stored in the variables A and B.

Answers displayed on their own are not very useful. If you add these PRINT lines to the program, you can make the computer display the sum next to the answer.

## Cake program

```
10 CLS
20 LET C=12
30 LET M=4
40 LET A=C/M
50 PRINT "THERE ARE ";A;" CAKES"
60 PRINT "FOR EACH MONSTER"
```

Run the program to find out how many cakes each monster will get.

Four monsters have been given 12 cakes and they want to share them out equally. The program above will solve the monsters'

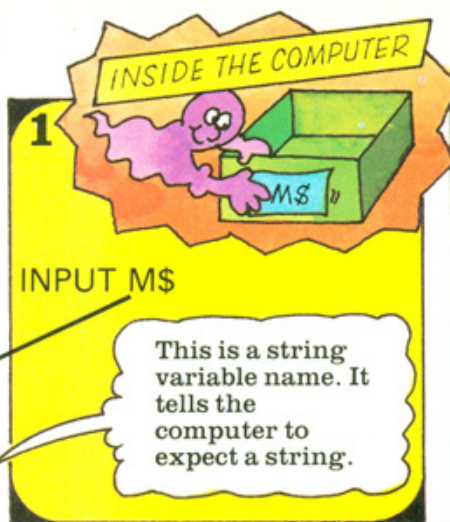
problem. The variable C contains the number of cakes. M is the number of monsters and A is the answer.



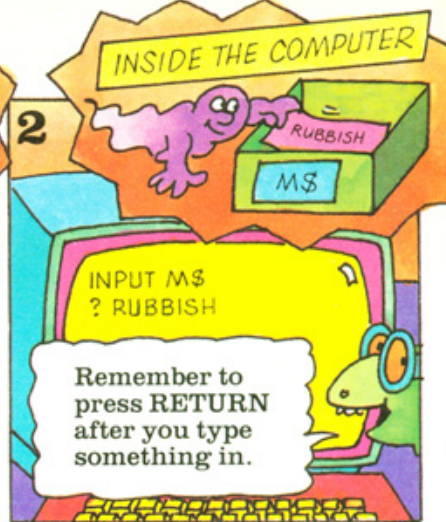


# Using INPUT

Another way to give the computer information is with the command INPUT. INPUT lets you put information into the computer's memory while the program is running. This means that you can give the computer different information, each time you run a program. You can find out how INPUT works on the right.



You always put a variable name after INPUT. When the computer meets INPUT it labels a memory space and waits for you to type something.



The computer tells you it is waiting by displaying a question mark. When you type the information the computer stores it in the variable.

## A program with INPUT

The PRINT commands display the questions.

```
10 CLS
20 PRINT "WHAT'S YOUR NAME?"
30 INPUT N$
40 PRINT "HOW OLD ARE YOU?"
50 INPUT A
60 CLS
70 PRINT N$;" IS ";A
80 PRINT "YEARS OLD"
```

This line tells the computer to expect words.

This line tells the computer to expect a number.

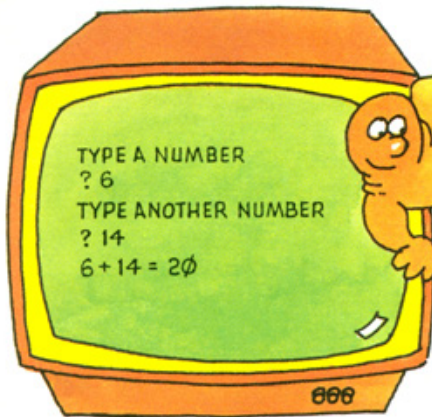


Here is a program to try. It uses INPUT to let you type in your name and age. Then the computer displays this information on the

screen. You can run the program lots of times, giving different names and ages each time. Try some silly ones.



## Computer calculator



You need to store the numbers in two variables called A and B.

This line tells the computer to add the two numbers and store the answer in C.

```
10 CLS
20 PRINT "TYPE A NUMBER"
30
40 PRINT "TYPE ANOTHER NUMBER"
50
60 LET C=A+B
70 PRINT A;"+";B;"=";C
```

See if you can fill in the missing lines in this program. The program makes your computer work as a calculator. You type in two numbers and the computer adds them

together and puts the answer on the screen. You need to fill in lines 30 and 50 with INPUT commands. If you get stuck, you can look at the answer on page 44.

## Message program puzzle

```
10 CLS
20 PRINT "WHAT IS THE MESSAGE?"
30 INPUT M$
40 PRINT "WHO IS IT FOR?"
50 INPUT A$
60 PRINT "WHO IS IT FROM?"
70 INPUT B$
80 CLS
90 PRINT
100 PRINT M$
110 PRINT
```

These are the commands you need to complete. You can check your answer on page 44.

Here is another program puzzle. When you run the program the computer asks for a message. Then it asks who the message is for and who is sending it.

The computer stores this information in variables A\$, B\$ and M\$. Then it clears the screen and displays the message.

Can you complete the PRINT commands in lines 90 and 110 so the computer displays the names stored in A\$ and B\$, with the message?



# Fun with PRINT

On these two pages you can find out how to display things in different places on the screen using PRINT.

## PRINT TAB



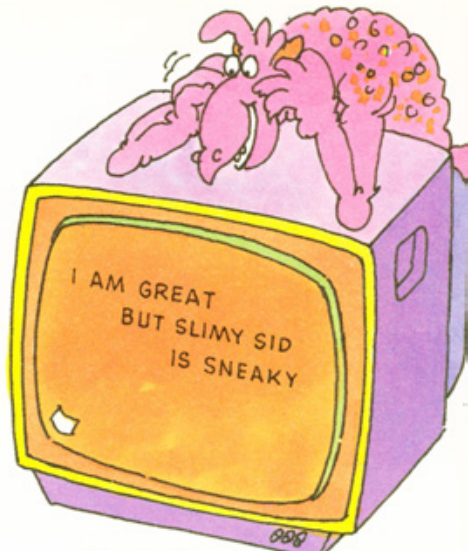
Try changing the number after TAB and see what happens.

```
10 CLS
20 PRINT TAB(5);"I AM GREAT"
30 PRINT TAB(8);"BUT SLIMY SID"
40 PRINT TAB(12);"IS SNEAKY"
```



The command PRINT TAB tells the computer to leave some spaces before a word or

number on the screen. You tell the computer how many spaces to leave by putting a



number in brackets after TAB. To see how this works, try running the program.

## PRINT TAB program



```
10 CLS
20 PRINT "HOW MANY SPACES"
30 PRINT "DO YOU WANT TO LEAVE?"
40 INPUT N
50 PRINT TAB(N);"HELLO"
```

In this program you tell the computer how many spaces to leave before the word HELLO. The computer stores the number in the variable N,

and then uses N with the PRINT TAB command. Run the program several times and give the computer a different number each time.

## Leaving empty lines

```
10 CLS
20 PRINT "HELLO"
30 PRINT
40 PRINT
50 PRINT
60 PRINT "GOODBYE"
```

To make the computer leave empty lines on the screen you use PRINT by itself. This program makes the computer leave three empty lines between the two messages.



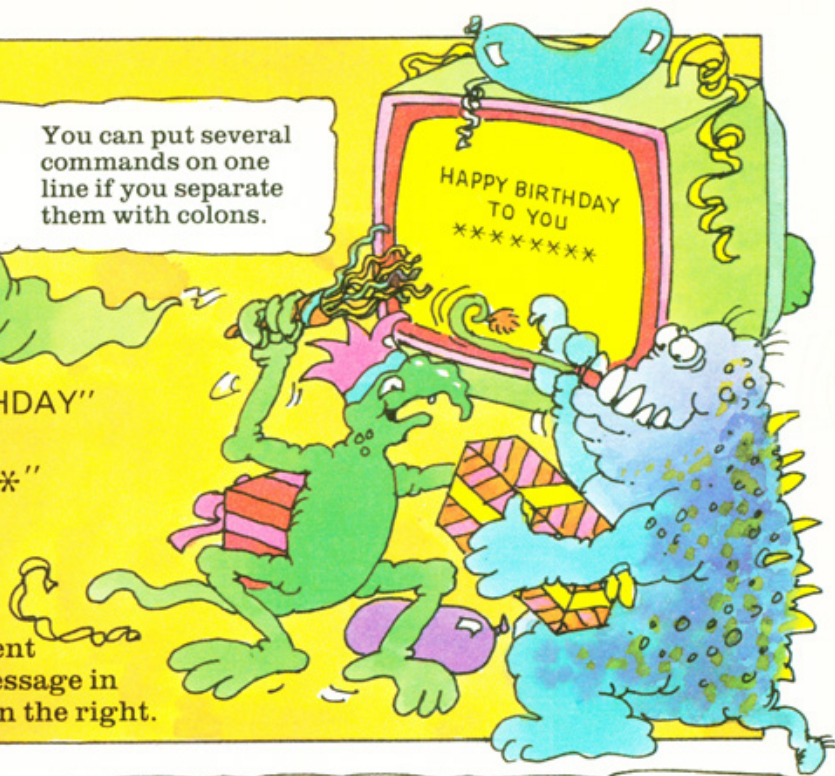
## Birthday program puzzle

This is a colon.

You can put several commands on one line if you separate them with colons.

```
10 CLS
20 PRINT:PRINT:PRINT
30 PRINT:PRINT:PRINT
40 PRINT TAB( );"HAPPY BIRTHDAY"
50 PRINT TAB( );"TO YOU"
60 PRINT TAB( );"*****"
```

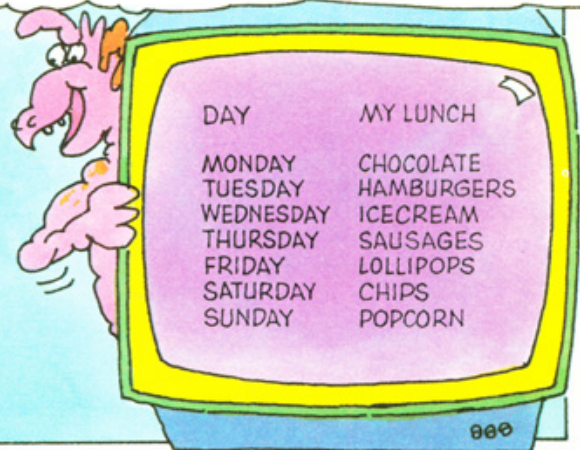
This program displays a birthday message. Before you run it, you must put the numbers in the brackets after the PRINT TAB commands. Experiment with different numbers to see if you can put the message in the middle of the screen, as shown on the right.



## Making columns

```
10 CLS
20 PRINT:PRINT:PRINT
30 PRINT "DAY","MY LUNCH"
40 PRINT
50 PRINT "MONDAY","CHOCOLATE"
60 PRINT "TUESDAY","HAMBURGERS"
70 PRINT "WEDNESDAY","ICECREAM"
80 PRINT "THURSDAY","SAUSAGES"
90 PRINT "FRIDAY","LOLLIPOPS"
100 PRINT "SATURDAY","CHIPS"
110 PRINT "SUNDAY","POPCORN"
```

Try running the program. This is what your screen should look like.



To make the computer display words or numbers in columns, you use commas.

A comma tells the computer to put the next word in a new column.

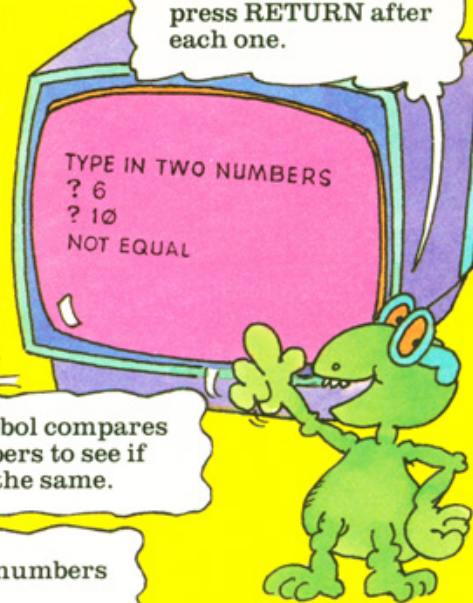


# Comparing things

You can make a computer compare pieces of information and then do different things according to the results. To do this you use the commands IF and THEN. You can find out how IF and THEN work in the program below.

```
10 CLS
20 PRINT "TYPE IN TWO NUMBERS"
30 INPUT A
40 INPUT Z
50 IF A=Z THEN PRINT "EQUAL"
60 IF A<>Z THEN PRINT "NOT EQUAL"
```

Type in the numbers one at a time and press RETURN after each one.



This symbol compares the numbers to see if they are the same.

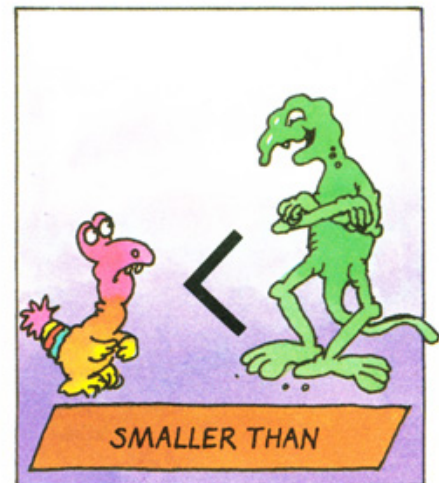
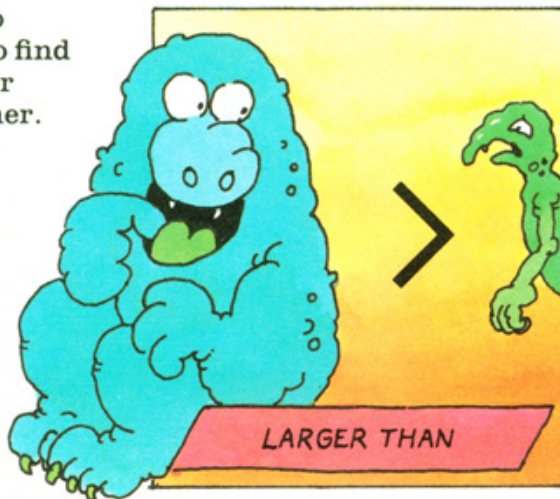
This symbol compares the numbers to see if they are different.

In this program the IF and THEN commands make the computer compare the numbers stored in the variables A and Z. If the

numbers are the same the computer displays the message "EQUAL". If the numbers are different, the message is "NOT EQUAL".

## More ways to compare things

A computer can also compare numbers to find out if one is larger or smaller than the other. The pictures on the right show the symbols for doing this.





## Comparing words

You can also use IF and THEN to make the computer compare words. Try running the program below. The IF and THEN commands make the computer compare the names stored in S\$ and N\$, to see if they are the same or not.

You can put your own name here if you like.

```
10 CLS
20 LET S$="SLIMY SID"
30 PRINT "WHAT'S YOUR NAME?"
40 INPUT N$
50 IF N$<>S$ THEN PRINT "I CAN'T STAND MONSTERS CALLED ";N$
60 IF N$=S$ THEN PRINT "HELLO ";S$;" NICE TO MEET YOU"
```

On some computers this message is too long to fit on one line so it will appear on two lines like this.

WHAT'S YOUR NAME  
? PINKY  
I CAN'T STAND MONSTERS  
CALLED PINKY

## Age program

```
10 CLS
20 LET A=11
30 PRINT "I AM ";A;" YEARS OLD"
40 PRINT "HOW OLD ARE YOU?"
50 INPUT B
60 IF A=B THEN PRINT "WE'RE THE SAME AGE!"
70 IF A>B THEN PRINT "I'M OLDER THAN YOU"
80 IF A<B THEN PRINT "I'M YOUNGER THAN YOU"
```

Put your own age here instead of 11.

I AM 11 YEARS OLD  
HOW OLD ARE YOU ?  
? 10  
I'M OLDER THAN YOU

When you type in this program, put your own age in line 20. Then let a friend run the program. The computer will work out

whether your friend is older, younger or the same age as you.



# IF and THEN programs

You can make a computer do all sorts of things after an IF and THEN command. For instance, you can tell it to jump to another program line using the command GOTO. You can also make it stop the program.

## Using GOTO

```
10 CLS
20 LET C=17
30 PRINT "GUESS HOW MANY CAKES"
40 PRINT "SLIMY SID HAS EATEN?"
50 INPUT G
60 IF G<>C THEN GOTO 50
70 PRINT "YOU'VE GUESSED IT!"
```

This line sends the computer back to line 50 if your guess is wrong.

The computer carries out this line if your guess is correct.

In this program you have to guess how many cakes Slimy Sid has eaten. If your guess is

wrong, the GOTO command makes the computer jump back to line 50 so you can have

another go. The computer will go on doing this until you get the answer right.

## Improving the program

The cake program would be clearer if the computer told you when your guess was wrong. You can do this on most computers, by putting a PRINT command after THEN in line 60.

Try typing in this version of line 60. See what difference it makes to the program.

```
60 IF G<>C THEN PRINT "WRONG":GOTO 50
```

Both PRINT and GOTO must be on the same line as IF and THEN. You separate them with a colon.



## IF and THEN with STOP

1

```
10 CLS
20 LET P$="BAKED BEANS"
30 PRINT "WHAT IS THE PASSWORD?"
40 INPUT A$
50 IF A$<>P$ THEN STOP
60 PRINT "CORRECT"
70 PRINT "WELCOME, FRIEND"
```

The computer will only carry out these lines if you type the correct password.

This is the password. It is stored in the variable P\$.

2

```
WHAT IS THE PASSWORD?
? CREAM CAKES
```

```
STOP AT LINE 50
```

This message tells you the program has stopped.

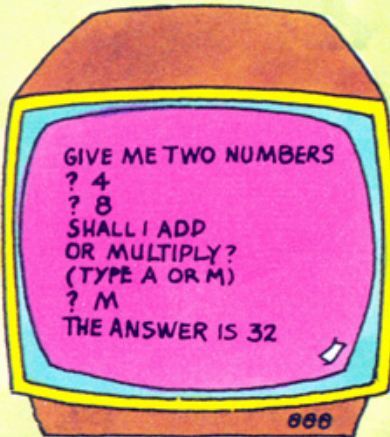
This is not the right password.

This password program uses the command STOP after IF and THEN. It tells the computer to stop the

program if you type in the wrong password. Most computers display a message

when they stop a program. This lets you know what line the program has stopped at.

## Maths machine



Try this program. When you run it, you give the computer two numbers. Then you tell it whether to add or multiply

```
10 CLS
20 PRINT "GIVE ME TWO NUMBERS"
30 INPUT X:INPUT Y
40 PRINT "SHALL I ADD"
50 PRINT "OR MULTIPLY?"
60 PRINT "(TYPE A OR M)"
70 INPUT R$
80 IF R$="A" THEN LET Z=X+Y
90 IF R$="M" THEN LET Z=X*Y
100 PRINT "THE ANSWER IS ";Z
```

Remember, you can put two commands on one line if you separate them with a colon.

The answer is stored in Z.

them by typing A or M. Lines 80 and 90 use IF and THEN to make the computer do the calculation you asked for.

The LET commands tells the computer to work out the sum and store the answer in Z.



# Making loops

A loop makes a computer do something, such as display a message, over and over again. To make a loop in a program, you use the commands FOR, TO and NEXT.

This is a loop program. It makes the computer display a message five times.

You can use a loop to repeat a message like this.

```
HELLO UGLYMUG  
HELLO UGLYMUG  
HELLO UGLYMUG  
HELLO UGLYMUG  
HELLO UGLYMUG
```

## Uglymug loop

```
10 CLS  
20 FOR K=1 TO 5  
30 PRINT "HELLO UGLYMUG"  
40 NEXT K
```

This is the PRINT command inside the loop.

This is the beginning of the loop.

This is the end of the loop.

The loop in this program begins with the FOR, TO command in line 20. It tells the computer how many

times to repeat the PRINT command. The command, NEXT, in line 40 tells the

computer to loop back and carry out the PRINT command again.

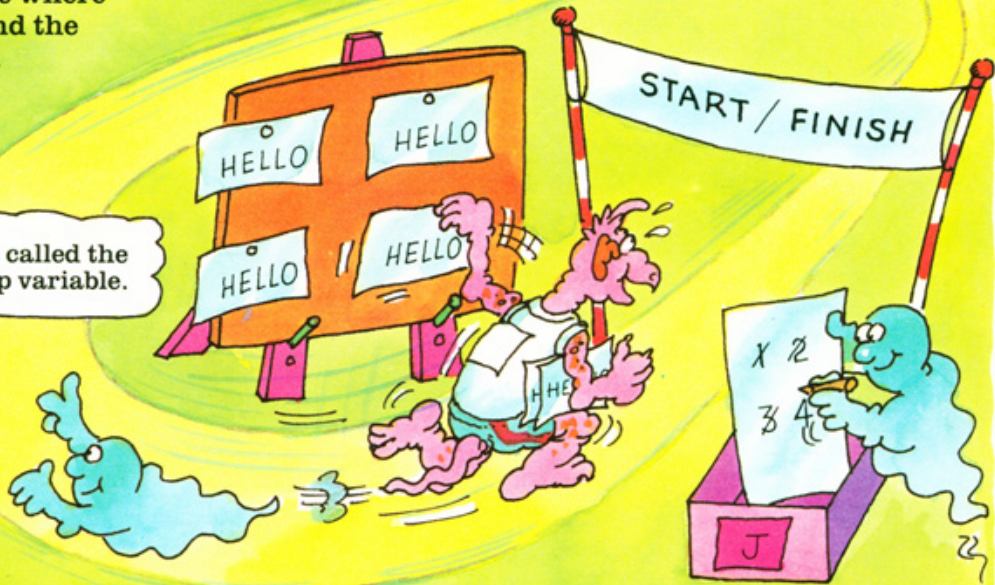


## How a loop works

A loop is rather like a race where the runner has to go round the track several times. Each time the computer carries out the instruction in a loop, it is like running one lap of the race.

J is called the loop variable.

```
10 CLS
20 FOR J=1 TO 6
30 PRINT "HELLO"
40 NEXT J
```



In this loop program, the computer uses the variable J to keep count of the laps. At

the beginning, the number in J is 1. Each time the computer goes round the

loop, the number in J goes up by 1 until it reaches 6. Then the loop is finished.

## Looking inside a loop

```
10 CLS
20 FOR L=1 TO 20
30 PRINT "LAP ";L
40 NEXT L
```



This program lets you look inside a loop, so you can see the computer keeping count of the laps.

## Loop puzzle

```
10 CLS
20 FOR J= TO
30 PRINT
40 NEXT J
```

Type the missing numbers in this line.

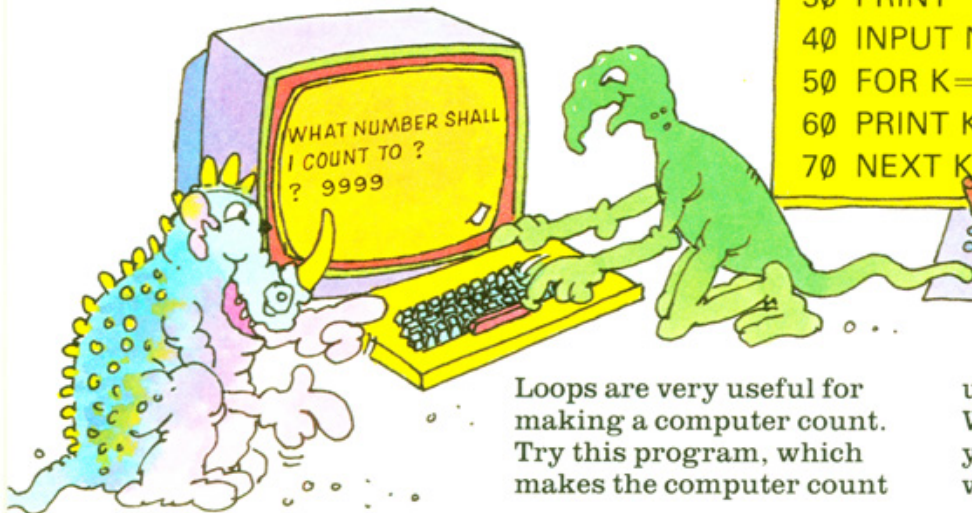
Type your name here, in quotes.

Can you complete this program, so the computer displays your name ten times. You need to fill in the numbers in line 20, and put your name in quotes after the PRINT command in line 30.



# Loopy programs

Here are some more ways of using loops in programs.



```
10 CLS          Counting program
20 PRINT "WHAT NUMBER SHALL"
30 PRINT "I COUNT TO?"
40 INPUT N
50 FOR K=1 TO N
60 PRINT K
70 NEXT K
```

Loops are very useful for making a computer count. Try this program, which makes the computer count

up to any number you like. When you run the program, you type in the number you want the computer to count to.

## Different ways of counting

This line tells the computer to count from 0 to 30 in steps of 5.

```
10 CLS
20 FOR L=0 TO 30 STEP 5
30 PRINT L
40 NEXT L
```

0  
5  
10  
15  
20  
25  
30

Try changing the number after STEP.

You can change the way the computer counts. For instance, you can make it count in jumps. To do this, you use the command

STEP followed by a number. The program above makes the computer count in steps of 5. Try running it.

## Countdown

```
10 CLS
20 PRINT "COUNTDOWN"
30 FOR J=10 TO 1 STEP -1
40 PRINT J
50 NEXT J
60 PRINT "BLAST OFF!"
```

STEP with a minus number makes the computer count backwards. Try this countdown program to see how it works.



## Making the computer wait

```
1 10 CLS
  20 PRINT "HELLO EVERYBODY"
  30 FOR K=1 TO 1000
  40 NEXT K
  50 PRINT "HELLO AGAIN"
```

This is the delay loop. It makes the computer count from 1 to 1000 before it carries on with the rest of the program.

A loop with no instructions inside it makes the computer pause for a moment. It is called a delay loop. In the program above, the delay loop in lines 30 and 40 makes the computer pause before it displays the second message.

```
2 30 FOR K=1 TO 500
  30 FOR K=1 TO 2000
```

A smaller number makes a shorter pause.

A larger number makes a longer pause.

You can alter the length of the pause by changing the second number in the FOR, TO command. Have a go at shortening or lengthening the pause in the HELLO program.

## Using the loop variable

```
1 Each time the
  computer goes round
  the loop, it multiplies
  9 by the number in K,
  and displays the answer.
```

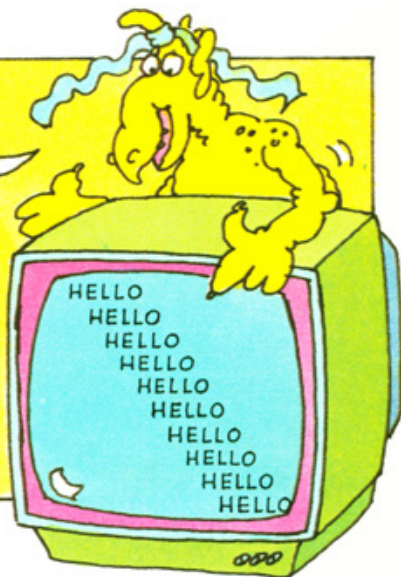
```
10 CLS
20 FOR K=1 TO 12
30 LET A=K*9
40 PRINT K;" x 9 =";A
50 NEXT K
```

This program makes the computer display the nine-times table. The loop variable, K, is used both for counting the loops, and for doing the sum inside the loop.

```
2 This is what the
  screen should look
  like when you run the
  program.
```

```
10 CLS
20 FOR J=1 TO 10
30 PRINT TAB(J);"HELLO"
40 NEXT J
```

In this program the computer uses the loop variable, J, as the number for the PRINT TAB command. This means that each time the computer repeats the loop it leaves one more space before the word HELLO.





# Random numbers

When you throw dice it is hard to guess, or predict, what numbers you will get. Unpredictable numbers like these are called random numbers.

On the right you can find out how to make a computer produce random numbers. The command is quite complicated, but it is useful in all sorts of programs.



**1**

Look at page 33 to find your computer's RND command.

INT(RND(1) )

You can find out what numbers to put here, in the box on the right.

To make a computer produce a random number you use the commands INT and RND(1) (or RND or RND(0) on some computers) followed by some numbers.

## Displaying random numbers

```
10 CLS
20 PRINT "HERE IS A NUMBER"
30 PRINT "BETWEEN 1 AND 20"
40 PRINT INT(RND(1)*20+1)
```

There are 20 numbers to choose from, starting with 1.

Remember to change this to suit your computer, if necessary.

HERE IS A NUMBER  
BETWEEN 1 AND 20  
17

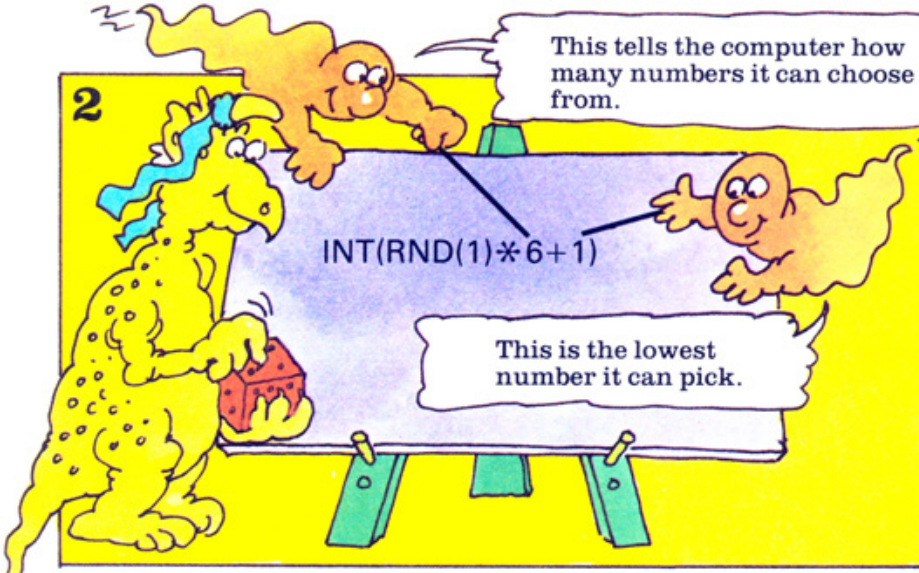
The computer may pick the same number twice in a row sometimes.

Here is a simple random number program to try. Line 40 tells the computer to pick

a number between 1 and 20 and display it on the screen. Run the program a few

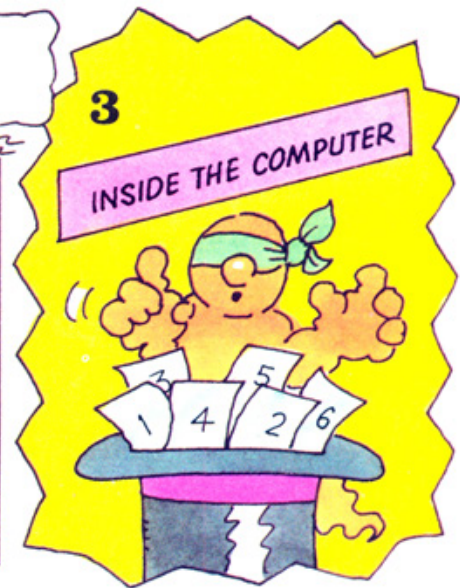
times. The computer will pick a new number each time.





To complete the instruction you need to multiply by the number of numbers the computer can choose from, and add the lowest number it can pick. For example, to

make the computer pick numbers from 1 to 6, like a dice, you need the instruction above. Make sure you use your computer's RND command.



If you give a computer the random number command shown in box 2, it picks a number between 1 and 6, as if it were blindfold.

### Lucky draw program



The monster gang are having a lucky draw. They have sold 30 tickets, numbered from 1 to 30, and they want the computer to pick the winning number. The program on the right will make the computer do this, but it is not

Remember to change this if you need to.

You should fill in the missing numbers here.

```

10 CLS
20 LET W=INT(RND(1)*  + )
30 PRINT "HELLO MONSTERS"
40 PRINT "THE LUCKY WINNER"
50 PRINT "IS NUMBER ";W

```

complete. See if you can fill in the missing numbers in line 20. This is the line that tells the computer to pick a random number between 1 and 30. The answer is on page 45.



# Debugging programs

Debugging a program means correcting the mistakes in it. Below you can find out how to debug programs.

At the bottom of the page, there are some hints for spotting bugs.

## 1 SYNTAX ERROR IN LINE 20

Syntax error means that there is something wrong with a BASIC command.



Your computer may display an error message to help you find the bug. Look in your computer manual to find out what the error message means.

2 LIST  
10 CLS  
20 PRUNT "HELLO"  
30 PRINT "MONSTER"

Can you spot the bug in this program?



To find a bug, type LIST and press RETURN. This puts the program lines on the screen. Check each program line carefully.

3 The PRINT command is wrongly spelt.

LIST  
10 CLS  
20 PRUNT "HELLO"  
30 PRINT "MONSTER"

20 PRINT "HELLO"

Line retyped correctly.



When you spot a mistake, correct it by typing the whole line again, starting with its line number.

## Hints for spotting bugs

Mistakes are quite easy to correct, but they are often tricky to spot. Even the smallest bug can stop the program working. On the right there are some hints for what to look out for.

This program contains some common bugs.

Letter key used instead of number key.

```
10 PRINT WHAT'S YOUR NAME?  
20 INOUT A$  
30 PRINT "HELLO" A$  
40 PRINT "GOODBYE "; A$
```

BASIC command wrongly spelt.

Missing quotes

Missing semicolon



# Command chart

The chart below shows the commands that different computers use to clear the screen and pick random numbers. At

the bottom of the page you can find out the names for the RETURN and DELETE keys on different computers.

	Clear screen	Random number
<b>BBC and Electron</b>	CLS	INT(RND(1)* + )
<b>VIC 20 and Commodore 64</b>	PRINT CHR\$(147)	INT(RND(1)* + )
<b>Apple</b>	HOME	INT(RND(1)* + )
<b>Spectrum</b>	CLS	INT(RND* + )

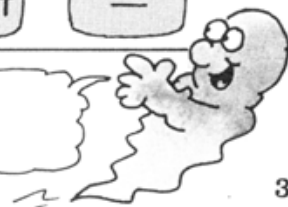
You need to fill in the numbers in this command as shown on page 31.



## Key chart

	RETURN	DELETE
<b>BBC and Electron</b>	RETURN	DELETE
<b>VIC 20 and Commodore 64</b>	RETURN	INST DEL
<b>Apple</b>	RETURN	←
<b>Spectrum</b>	ENTER	CAPS SHIFT    ⓪ _

Press both keys together.





# Programs and puzzles

Here and on the next few pages, there are programs to try and puzzles to solve. They all involve BASIC commands

covered in this book. If you get stuck with any of the puzzles, look at the answers on pages 45-47.

## 1 Words and spaces

```
10 CLS
20 PRINT "RED";"ROSE"
30 PRINT "BLACK";"COAL"
40 PRINT "WHITE";"SNOW"
```

Try this program. It makes the computer display words next to each other on the screen.

## Star lines

```
1
10 CLS
20 FOR K=1 TO 10
30 PRINT TAB(3);"*"
40 NEXT K
```

Run this program to make your computer display a line of stars at the side of the screen.

## 2

```
RED ROSE
BLACK COAL
WHITE SNOW
```

See if you can change the PRINT lines to make spaces between the words, as shown above.

## 2

```
*
*
*
*
*
*
*
*
*
```

By changing the number after TAB, can you put the stars in the middle of the screen?

## 3

```
RED      ROSE
BLACK    COAL
WHITE    SNOW
```

Can you change the PRINT lines again so the computer arranges the words in two columns like this?

## 3

```
*****
```



Hint: you use a semicolon to make the computer stay on the same line.

Can you change the program again to make a line of stars across the screen, like this?



## Party invitation puzzle

```
10 CLS
20 PRINT "*****"
30 PRINT "PLEASE COME"
40 PRINT "TO MY"
50 PRINT "BIRTHDAY PARTY"
60 PRINT "DATE 19TH JULY"
70 PRINT "TIME 3.00PM"
80 PRINT "PLACE MY HOUSE"
90 PRINT "LOVE"
100 PRINT "UGLYMUG"
110 PRINT "*****"
```



If you like, put your own name on the invitation and change the information.

```
*****

PLEASE COME
TO MY
BIRTHDAY PARTY

DATE      19 JULY
TIME      3.00 PM
PLACE     MY HOUSE

LOVE
UGLYMUG

*****
```

This program displays a party invitation on the left of the screen. Can you improve it so that the invitation is centred on the screen, as shown on the right?

You will need to replace the PRINT commands with PRINT TAB and add some empty PRINT lines to make spaces.

4

```
*****
 *
 *
 *
 *
 *
 *
 *
 *
 *
 *
```

Now see if you can write a program with two loops to display stars in a T-shape.

5

```
10 CLS
20 FOR K=1 TO 10
30 PRINT TAB(K);"*"
40 NEXT K
```

This program uses PRINT TAB followed by a loop variable to draw a diagonal line of stars. Each time the

6

```
*
 *
 *
 *
 *
 *
 *
 *
 *
```

computer repeats the loop a star is displayed one space further from the left of the screen.



## Calculation programs

The programs on these two pages make the computer do calculations. Try them and then see if you can do the puzzles.

### TV time

```
10 CLS
20 PRINT "HOW MANY HOURS DO YOU"
30 PRINT "SPEND WATCHING TV"
40 PRINT "EACH WEEK?"
50 INPUT X
60 LET Y=X*52
70 PRINT "THAT MEANS YOU SPEND"
80 PRINT Y;" HOURS WATCHING TV"
90 PRINT "EACH YEAR!"
```

```
HOW MANY HOURS DO YOU
SPEND WATCHING TV
EACH WEEK?
? 6
THAT MEANS YOU SPEND
312 HOURS WATCHING TV
EACH YEAR!
```

This program works out the number of hours you spend watching TV each year. When you run it you type the number of hours you watch each week. The computer multiplies this by 52 (the number of weeks in a year). Line 60 does the calculation.

### Calculation puzzles

**1**

```
HOW MANY ICE-CREAMS
DO YOU EAT
EACH WEEK?
? 2
THAT MEANS YOU EAT
104 ICE-CREAMS
EACH YEAR!
```

Can you adapt the TV time program to work out how many ice-creams you eat each year?

**2**

```
HOW MANY SWEETS DO YOU
EAT EACH DAY?
? 2
THAT MEANS YOU EAT
730 SWEETS
EACH YEAR!
```



You will need to change the calculation in line 60.

Now try changing the program again to make it calculate how many sweets you eat in a year, from the number you eat each day, as shown above.



## Averages program

You need to make the computer ask for your journey time in minutes.

```
10 CLS
20 PRINT "HOW MANY AGES DO"
30 PRINT "YOU WANT TO AVERAGE?"
40 INPUT N
50 LET T=0
60 FOR K=1 TO N
70 PRINT "WHAT IS AGE ";K;"?"
80 INPUT X
90 LET T=T+X
100 NEXT K
110 LET A=T/N
120 PRINT "THE AVERAGE IS ";A
```



## Puzzle

```
HOW MANY JOURNEYS DO
YOU WANT TO AVERAGE?
? 5
TYPE TIMES IN MINUTES
JOURNEY 1
? 21
JOURNEY 2
? 25
JOURNEY 3
? 31
JOURNEY 4
? 19
JOURNEY 5
? 29
YOUR AVERAGE JOURNEY
TIME WAS 25 MINUTES
```

The program above works out the average age of a group of people. Use it to find the average age of your family, or a group of friends. When you run the program you type the number of ages

you want to average and then the ages themselves. As you type each age, the computer adds them, to get the total age (T). It divides T by the number of ages (N) to get the average.

You can easily alter the program on the left to work out different averages. Try changing it to work out the average time it takes to get to school.

## Sums

```
1
10 CLS
20 PRINT "TYPE TWO"
30 PRINT "NUMBERS"
40 INPUT A
50 INPUT B
60 PRINT A;"x";B;"=";
70 PRINT A x B
```

This program makes the computer multiply any two numbers – but there is a bug in it. Can you spot the bug?

**2**

```
TYPE TWO NUMBERS
? 4
? 2
4 + 2 = 6
4 - 2 = 2
4 x 2 = 8
```

Try writing a program to make the computer add, subtract and multiply any two numbers you give it.

**3**

```
TYPE A NUMBER
? 72
72 x 5 = 360
```

Can you write a program to make the computer multiply any number by 5?



## Loop puzzles

Try these programs and puzzles to practise using loops.

### Joke program

```
10 CLS
20 PRINT "WHAT'S GREEN AND HAIRY"
30 PRINT "AND GOES UP AND DOWN?"
40 PRINT:PRINT:PRINT
50 FOR K=1 TO 2000:NEXT K
60 PRINT "A GOOSEBERRY"
70 PRINT "IN A LIFT!!"
```

WHAT'S GREEN AND HAIRY  
AND GOES UP AND DOWN?

A GOOSEBERRY  
IN A LIFT!!

You can put both parts  
of a loop on one line. You  
separate the two  
commands with a colon.

In this joke program the  
delay loop in line 50 makes

the computer pause before  
displaying the punchline.

Try running the program  
to see how it works.

### Delay loop puzzles

1

```
10 CLS
20 FOR K=1 TO 100
30 PRINT "HELLO"
40 FOR L=1 TO 20:NEXT L
50 CLS
60 FOR L=1 TO 20:NEXT L
70 NEXT K
```

These are delay loops.



This program puts a flashing message on the screen, but the message flashes too quickly and you cannot read it properly. Can you solve this problem?

2

```
10 CLS
20 FOR L=1 TO 60
30 PRINT L
40 FOR J=1 TO 50:NEXT J
50 NEXT L
```



Use a watch or clock to  
help you.

In this program the computer counts from 1 to 60. Try changing the size of the delay loop so the computer counts with pauses of one second, like a watch. You will need to experiment with different-sized loops.



## Number sequences

```
1 10 CLS
20 LET R=INT(RND(1)*10+1)
30 PRINT "HERE ARE THREE NUMBERS"
40 FOR J=2 TO 4
50 PRINT J*R
60 NEXT J
70 PRINT "WHAT IS THE NEXT"
80 PRINT "IN THE SEQUENCE?"
90 LET N=5*R
100 INPUT X
110 IF X=N THEN PRINT "YES, WELL DONE"
120 IF X<>N THEN PRINT "NO, IT'S ";N
```

The program above is for a number game. It uses a random number command and a loop to produce a

sequence of three numbers. You have to guess the next number in the sequence.

2

```
HERE ARE THREE NUMBERS
6
9
12
WHAT IS THE NEXT
IN THE SEQUENCE?
? 15
YES, WELL DONE
```



Hint: you will need a STEP loop with a minus number.

This is what your screen looks like when you run the program.

3

```
HERE ARE THREE NUMBERS
24
18
12
WHAT IS THE NEXT
IN THE SEQUENCE?
? 6
YES, WELL DONE
```

Can you change the game so the sequence of numbers goes backwards? You need to alter the loop and the calculation in line 90.

## Counting puzzle

```
0
100
200
300
400
500
```

Can you write a program to make the computer count from 0 to 500 in jumps of 100?

## Screen puzzle

Hint: you need to put a PRINT command inside a loop.



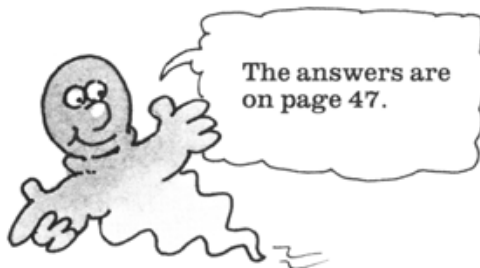
THIS IS THE BOTTOM OF THE SCREEN!

See if you can make the computer display a message like this at the bottom of the screen.



## Number puzzles

Try the puzzles on the right. They will help you to practise the random number command on your computer. Then see if you can fill in the missing random number commands in the games programs below.



**1**

THE RANDOM NUMBER  
IS 18

Can you write a program to make the computer pick a random number between 10 and 20, and display it on the screen, like this?

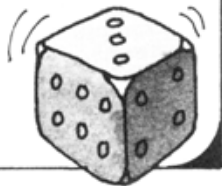
**2**

HERE ARE FIVE NUMBERS  
10  
13  
19  
17  
14

Now try writing a program to make the computer display five random numbers.

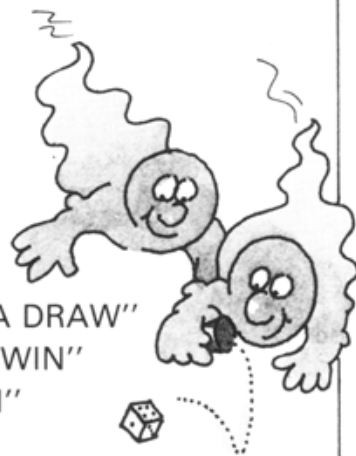
## Dice game

```
YOUR THROW  
? 5  
MY THROW  
3  
YOU WIN
```



To play this game you need a dice. You throw the dice and type in the number you get.

```
10 CLS  
20 PRINT "YOUR THROW"  
30 INPUT X  
40 PRINT "MY THROW"  
50 LET R=  
60 PRINT R  
70 IF X=R THEN PRINT "IT'S A DRAW"  
80 IF X>R THEN PRINT "YOU WIN"  
90 IF X<R THEN PRINT "I WIN"
```



Then the computer "throws" by picking a number between 1 and 6. Before the program

will work, you must put the random number command in line 50.

## Number guessing game

```
10 CLS
20 LET R=
30 PRINT "I'M THINKING OF A"
40 PRINT "NUMBER BETWEEN"
50 PRINT "1 AND 25."
60 PRINT "GUESS WHAT IT IS."
70 INPUT G
80 IF G=R THEN GOTO 120
90 IF G<R THEN PRINT "TOO SMALL"
100 IF G>R THEN PRINT "TOO BIG"
110 GOTO 70
120 PRINT "YES, THAT'S IT"
```

```
I'M THINKING OF A
NUMBER BETWEEN
1 AND 25.
GUESS WHAT IT IS.
? 4
TOO SMALL
? 6
TOO BIG
? 5
YES, THAT'S IT.
```

In this game the computer chooses a number and you guess what it is. Each time you guess, the computer tells you whether your number is

too big or too small. The game does not end until you guess the number correctly. Before you run the program,

put a random number command in line 20 to make the computer pick a number between 1 and 25.

## Puzzle

1

```
I'M THINKING OF A
NUMBER BETWEEN
1 AND 25.
GUESS WHAT IT IS.
? 8
TOO BIG
? 7
TOO BIG
? 3
YES, THAT'S IT.
YOU MADE 3 GUESSES.
```

2

```
15 LET C=0
```

You need to work out what line numbers the extra lines should have.



Can you improve the game so the computer keeps count of the number of guesses you make? You will need a new variable C. First you add line 15, shown above, to store the

number 0 in C. Then you need a line which adds 1 to the number in C each time you make a guess. Finally, add a PRINT line to display the number of guesses you made.



## Spot the bug puzzles

There are deliberate mistakes in each of the programs on these two pages. Some are bugs which will stop the program running. Others just make the program do silly things.

See if you can spot the mistakes and correct them so the programs run properly.

### Computer counter

1

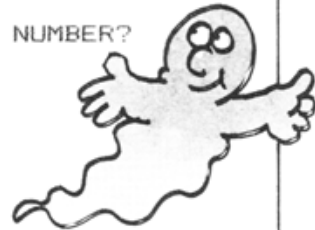
```
10 CLS
20 PRINT "FIRST NUMBER?"
30 INPUT A
40 PRINT "LAST NUMBER?"
50 INPUT B
60 FOR K=B TO A
70 PRINT K
80 NEXT K
```



2

This is how the program works when it is debugged.

```
FIRST NUMBER?
? 12
LAST NUMBER?
? 15
12
13
14
15
```



The program on the left makes the computer count. You choose the number it starts counting from and the number it stops at. Can you spot the bug in the program?

### Times table machine

1

```
10 CLS
20 PRINT "TIMES TABLE MACHINE"
30 PRINT:PRINT
40 PRINT "WHAT TABLE DO YOU WANT?"
50 INPUT T
60 FOR J=1 TO 12
70 LET A=J*T
80 PRINT J;" x ";T;" = ";A
90 NEXT Z
```



2

```
TIMES TABLE MACHINE
WHAT TABLE DO YOU WANT?
? 6
1 x 6 = 6
2 x 6 = 12
3 x 6 = 18
4 x 6 = 24
5 x 6 = 30
6 x 6 = 36
7 x 6 = 42
8 x 6 = 48
9 x 6 = 54
10 x 6 = 60
11 x 6 = 66
12 x 6 = 72
```

If you run this times table program as it is, it will not work. Can you see what is

wrong with it? The computer's error message may help you spot the bug.

This is what the screen looks like when the program is working correctly.





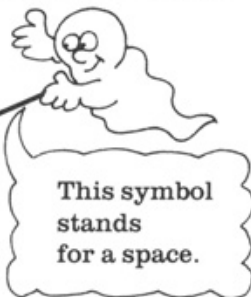
# Puzzle answers

To find the answer to a puzzle, look for the page number and name of the puzzle.

For some of the puzzles you may find that your program is slightly different from the one given. This does not matter, so long as your program runs correctly.

## Page 12 Face program

```
10 CLS
20 PRINT "□□□***"
30 PRINT "□□*****"
40 PRINT "□*****"
50 PRINT "□1□□□□□1"
60 PRINT "□1□□□□1"
70 PRINT "(□□□,□□□)"
80 PRINT "□1□□□□□1"
90 PRINT "□1□□□□□1"
100 PRINT "-----"
```



This program produces a face like the one shown.

## Page 14 LET puzzle

```
LET C=11
LET G=19
LET Z=30
```

These LET commands put the numbers into the variables.

## Page 15 PRINT puzzle

```
50 PRINT A$
60 PRINT B$
70 PRINT C$
```

To display the words stored in A\$, B\$ and C\$ you put PRINT commands followed by the variable names, as shown above.

## Page 19 Computer calculator

```
30 INPUT A
50 INPUT B
```

These are the commands for lines 30 and 50. They tell the computer to store the numbers you type in the variables A and B.

## Message program

```
90 PRINT A$
110 PRINT B$
```

To make the computer display the names stored in A\$ and B\$, put the variable names after the PRINT commands.

## Page 21 Birthday program puzzle

```
40 PRINT TAB(13);"HAPPY BIRTHDAY"
50 PRINT TAB(18);"TO YOU"
60 PRINT TAB(17);"*****"
```

To centre the words on the screen you put numbers in the brackets after TAB. The numbers vary from computer to computer. The ones shown above should work on most computers except the Spectrum and VIC 20. You need to use smaller numbers on these computers.

## Page 27 Loop puzzle

```
20 FOR J=1 TO 10
30 PRINT "DEADEYE DICK"
```

You should put the numbers 1 and 10 in line 20 and add your name to the PRINT command in line 30, as shown above.

## Page 31 Lucky draw program

```
20 LET W=INT(RND(1)*30+1)
```

This is the completed random number command. You may need to remove the (1) or put (0) after RND on your computer.

## Page 34 Words and spaces

```
2 20 PRINT "RED ";"ROSE"  
30 PRINT "BLACK ";"COAL"  
40 PRINT "WHITE ";" SNOW"
```

To make spaces between the words, you type a space inside the quotes on one side of the semi-colon.

```
3 20 PRINT "RED","ROSE"  
30 PRINT "BLACK","COAL"  
40 PRINT "WHITE","SNOW"
```

To put the words in columns, replace the semi-colons with commas. You do not need spaces inside the quotes.

## Star lines

2 To make the line of stars go down the middle of the screen, you put a bigger number after TAB. On most computers the number is 19. On the Spectrum it is 15 and on the VIC 20 it is 10.

```
3 30 PRINT "*" ;
```

To display the line of stars across the screen, remove the TAB command in line 30 and put a semi-colon after the star.

30 is the number of tickets sold.



It doesn't matter which side of the semi-colon you type the space.



```
4 10 CLS  
20 FOR K=1 TO 10  
30 PRINT "*" ;  
40 NEXT K  
50 FOR K=1 TO 10  
60 PRINT TAB(5); "*" ;  
70 NEXT K
```

This is the program to make a T-shape. It uses the loops in puzzles 2 and 3.

## Page 35 Party invitation puzzle

To centre the lines, you replace the PRINT commands with PRINT TAB. The numbers after TAB will vary depending on your computer.

## Page 36 Calculation puzzles

```
1 20 PRINT "HOW MANY ICE-CREAMS"  
30 PRINT "DO YOU EAT"  
70 PRINT "THAT MEANS YOU EAT"  
80 PRINT Y;" ICE-CREAMS"
```

You need to change the words in the PRINT lines as shown above.

```
2 20 PRINT "HOW MANY SWEETS DO YOU"  
30 PRINT "EAT EACH DAY?"  
60 LET Y=X*365  
70 PRINT "THAT MEANS YOU EAT"  
80 PRINT Y;" SWEETS"
```



365 is the number of days in the year.

Change the PRINT lines as shown above. You do not need line 40. To get rid of it, type 40 and press RETURN. In line 60, multiply X by 365.



## Page 37

### Averages puzzle

```
20 PRINT "HOW MANY JOURNEYS DO"  
30 PRINT "YOU WANT TO AVERAGE?"  
45 PRINT "TYPE TIMES IN MINUTES"  
70 PRINT "JOURNEY ";K  
120 PRINT "YOUR AVERAGE JOURNEY"  
130 PRINT "TIME WAS ";A;" MINUTES"
```

To alter the averages program you change the words in the PRINT lines. You should also add a PRINT command (line 45) to ask for the journey times in minutes.

### Sums

```
1 70 PRINT A*B
```

The sum in line 70 is not correct. You should use the computer's multiplication symbol, \*, instead of x.

```
2 10 CLS  
20 PRINT "TYPE TWO NUMBERS"  
30 INPUT X:INPUT Y  
40 PRINT X;" + ";Y;" = ";X+Y  
50 PRINT X;" - ";Y;" = ";X-Y  
60 PRINT X;" * ";Y;" = ";X*Y
```

This program makes the computer add, subtract and multiply any two numbers.

```
3 10 CLS  
20 PRINT "TYPE A NUMBER"  
30 INPUT N  
40 PRINT N;" * 5 = ";N*5
```

This program makes the computer multiply any number you give it by 5.

## Page 38

### Delay loop puzzles

1 You need to make the delay loops in lines 40 and 60, bigger. To do this, put a larger number after the command TO.

2

<b>BBC</b>	40 FOR J=1 TO 1540 :NEXT J
<b>Electron</b>	40 FOR J=1 TO 1130 :NEXT J
<b>VIC 20</b>	40 FOR J=1 TO 860 :NEXT J
<b>Commodore 64</b>	40 FOR J=1 TO 710 :NEXT J
<b>Apple</b>	40 FOR J=1 TO 820 :NEXT J
<b>Spectrum</b>	40 FOR J=1 TO 220 :NEXT J

The size of the loop varies from computer to computer. This chart gives the delay loops for a number of different computers.

## Page 39

### Number sequences

```
3 40 FOR J=4 TO 2 STEP-1  
90 LET N=1*R
```

The STEP loop in line 30 makes the sequence of numbers go backwards. The calculation in line 90 works out the next lowest number in the sequence.

### Counting puzzle

```
10 CLS  
20 FOR J=0 TO 500 STEP 100  
30 PRINT J  
40 NEXT J
```

You need a program with a STEP loop like this, to make the computer count from 0 to 500 in jumps of 100.

### Screen puzzle

```
10 CLS  
20 FOR K=1 TO 18  
30 PRINT  
40 NEXT K  
50 PRINT "THIS IS THE BOTTOM"  
60 PRINT "OF THE SCREEN!"
```

On some computers you may need to replace the figure 18 with 13.

This program makes the computer leave 18 empty lines so the message appears at the bottom of the screen.



## Page 40 Number puzzles

```
1  10 CLS
   20 LET R=INT(RND(1)*11+10)
   30 PRINT "THE RANDOM NUMBER"
   40 PRINT "IS ";R
```

You need a program like this to make the computer pick a number between 10 and 20 and display it. You may need to replace RND(1) with RND or RND(0) on your computer.

```
2  10 CLS
   20 PRINT "HERE ARE FIVE NUMBERS"
   30 FOR L=1 TO 5
   40 LET R=INT(RND(1)*11+10)
   50 PRINT R
   60 NEXT L
```

The random number and PRINT commands are inside a loop. Each time the computer goes round the loop, it picks another number and puts it on the screen.

## Dice game

```
50 LET R=INT(RND(1)*6+1)
```

You need to put this random number command in line 50. On some computers you may need to put RND or RND(0) instead of RND(1).

## Page 41 Number guessing game

```
20 LET R=INT(RND(1)*25+1)
```

This is the completed random number command. Remember, you may need to change RND(1) on some computers.

There are 11 numbers the computer can choose from between 10 and 20. The lowest is 10.



## Puzzle

```
75 LET C=C+1
130 PRINT "YOU MADE ";C;" GUESSES"
```

You need to add these two lines as well as line 15. Line 75 adds 1 to the number in C each time you make a guess. Line 130 displays the final number of guesses.

## Page 42 Computer counter

The variables A and B are the wrong way round in the FOR, TO command. Line 60 should look like this:

```
60 FOR K=A TO B
```

## Times table machine

The NEXT command is followed by the wrong variable. It should be followed by J, the loop variable.

```
90 NEXT J
```

## Page 43 Happy or sad face

When you type HAPPY, the computer displays a sad face and vice-versa. To correct this, swap the words HAPPY and SAD in the IF and THEN lines shown below.

```
80 IF A$="HAPPY" THEN PRINT "!000000!"
90 IF A$="SAD" THEN PRINT "!0-000-0!"
110 IF A$="HAPPY" THEN PRINT "!0:000:0!"
120 IF A$="SAD" THEN PRINT "!0000000!"
140 IF A$="HAPPY" THEN PRINT "!0000000!"
150 IF A$="SAD" THEN PRINT "!0:000:0!"
```



# Index

- addition, 13, 17, 19, 25
- averages, 37
  
- BASIC, 4, 6
- bugs, 11, 16, 42-43
  - how to correct, 32
  
- CLS, 7, 33
- colon, 21, 24, 38
- columns, 21, 34
  - how to make, 21
- comma, 21
- command, 4, 6
- comparing 22-23, 24-25
  - symbols for 22, 23
- computer
  - code, 5
  - languages, 4-5
- count, making computer,
  - 28, 39, 42
- cursor, 6
  
- debugging programs, 32
- delay loop, 29, 38
- DELETE, 10, 33
- division, 13, 17, 37
  
- ENTER, 6, 33
- error message, 11, 32
  
- FOR...NEXT, 26-27, 28-29
- FORTRAN, 5
- GOTO, 24
- HOME, 7, 33
  
- IF...THEN, 22-23, 24-25
- INPUT, 18-19
- INT, 30-31, 33
- LET, 14
- LIST, 11, 32
- LOGO, 5
- loops, 26-27, 28-29, 34, 35,
  - 38-39
  
- key chart, 33
  
- memory, 10, 14-15, 18
- multiplication, 13, 17, 25,
  - 36, 37, 39
  
- NEW, 11
  
- PRINT, 6-7, 12-13, 15, 16,
  - 17, 20-21, 34
- CHR\$ (147), 7, 33
- displaying messages
  - with, 20, 21, 35, 39
- making pictures with,
  - 12, 43
- TAB, 20-21, 29, 34-35
  
- programs, 4
  - how to alter, 11
  - how to type in and run,
    - 10, 11
  - writing, 8-9
  
- random numbers, 30-31,
  - 33, 39, 40-41
- RETURN, 6, 10, 33
- RND, 30-31, 33
- RUN, 10
  
- semicolon, 16
- STEP, 28, 39
- STOP, 25
- string, 15
- subtraction, 36
- sums, 13, 17, 25, 29, 36-37
  
- times tables, 29, 42
- turtle, 5
  
- variables, 14-15, 16-17,
  - 18-19, 41
  - loop, 27, 29, 35

First published in 1984 by  
Usborne Publishing Ltd,  
20 Garrick Street,  
London WC2E 9BJ,  
England.

All rights reserved. No part of this publication may be reproduced,  
stored in a retrieval system or transmitted in any form or by any  
means, electronic, mechanical, photocopying, recording or otherwise,  
without the prior permission of the publisher.

© 1984 Usborne Publishing Ltd.

The name Usborne and the  
device  are the Trade Marks  
of Usborne Publishing Ltd.





# FIRST COMPUTER LIBRARY

The First Computer Library is a lively and colourful new series for young beginners to computing. A gang of friendly ghosts and monsters show the reader what computers can do, how they work and how to program them. The books are packed with short, entertaining programs which will run on most makes of home computer.



## All About Computers

This colourful introduction explains in simple terms what a computer is, how it works and what it can do. Short programs are included to illustrate the explanations.



## Simple BASIC

A simple and fun introduction to programming in BASIC, with masses of entertaining programs to try and puzzles to solve.



## Computer Fun

This book is packed with programs to make a computer flash messages, play games, make pictures and lots besides. No previous programming knowledge is needed.

## The Usborne First Guide to the BBC Micro

This friendly guide for young readers explains exactly how to set up and use a BBC computer. A group of little monsters explain how to program in BBC BASIC, give helpful advice on things that can go wrong and show what the BBC micro can do.



ISBN 0-86020-801-X



9 780860 208013

ISBN 0 86020 801 X

£1.99