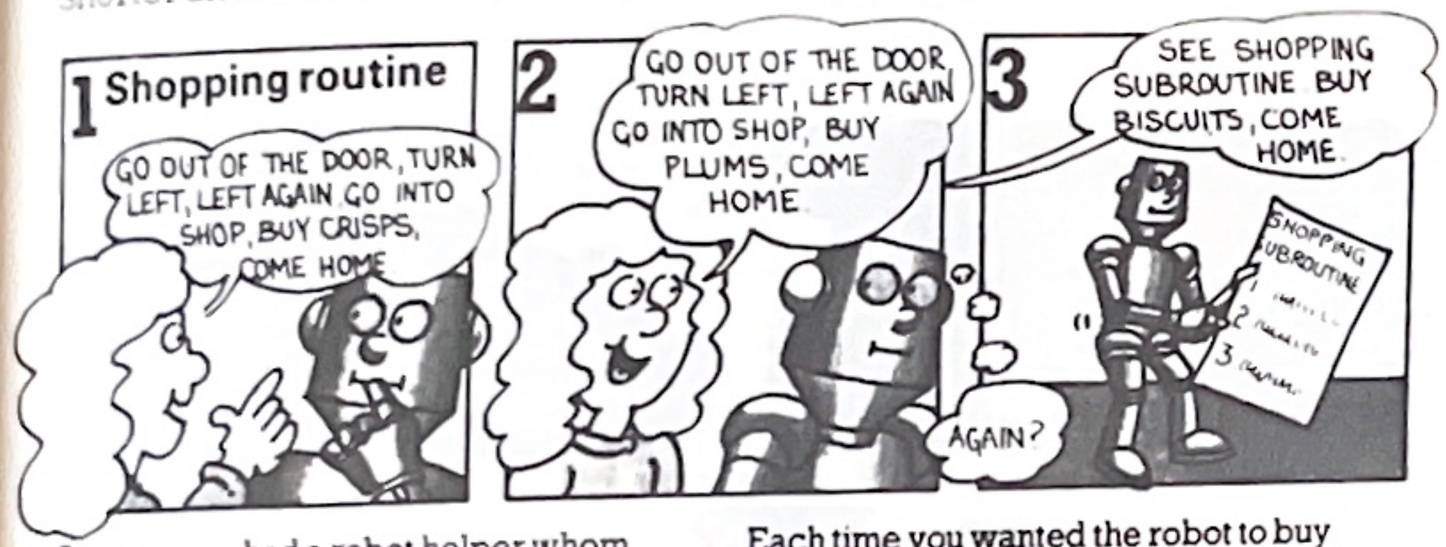
Subroutines

A subroutine is a sort of mini-program within a program. It carries out a particular task, such as adding numbers or keeping a score, and you can send the computer to it whenever you want this task carried out. This saves writing out the program lines each time and makes the program shorter and easier to read and type into the computer.



Suppose you had a robot helper whom you could program to run errands for you. If you wanted something from the shop you would have to give it precise instructions telling it how to get there.

Each time you wanted the robot to buy something you would have to give it the same instructions. It would be much simpler to give the robot a shopping subroutine and tell it to refer to it each time.

4 Shopping program

```
10 PRINT "WHAT DO YOU WANT FROM THE SHOP"
20 INPUT X$
30 GOSUB 100
40 PRINT "ANYTHING ELSE"
50 INPUT M$
60 IF M$="YES" THEN GOTO 10
70 STOP

100 REM: SHOP SUBROUTINE
110 PRINT "GO OUT, TURN LEFT"
120 PRINT "LEFT AGAIN, ENTER SHOP"
130 PRINT "BUY ";X$;" COME HOME"
140 RETURN
```

Line 30 sends the computer to the first line of the subroutine.

You need the word STOP at the end of the main program to stop the computer carrying on into the subroutine.

It is useful to label a subroutine with a REM line so you know what it is for.

This sends the computer back to line 40 – the line after GOSUB.

If you forget the RETURN line you get a bug.

In BASIC, to tell the computer to go to a subroutine you use the word GOSUB with the word RETURN at the end of the subroutine. GOSUB should be followed by the number of the first line of the subroutine. RETURN does not need a line number. The computer automatically goes back to the instruction after the one where it left the main part of the program. You can send the computer to a subroutine anywhere in the program as many times as you like.

Gosub programs

A subroutine is useful for carrying out any task which you want to repeat several times at different stages in the program. Here are some more programs with subroutines.

Numbers program

```
50 INPUT A
60 INPUT B
70 GOSUB 250
80 PRINT "A DIVIDED BY B=";A/B
90 GOTO 50
250 REM: SUBROUTINE TO STOP
260 IF A=0 AND B=0 THEN STOP
270 RETURN
```

This subroutine provides an escape from the program. If you want to stop dividing you input 0 at lines 50 and 60. This program does not need STOP before the subroutine as line 80 sends it back.

Conversion program

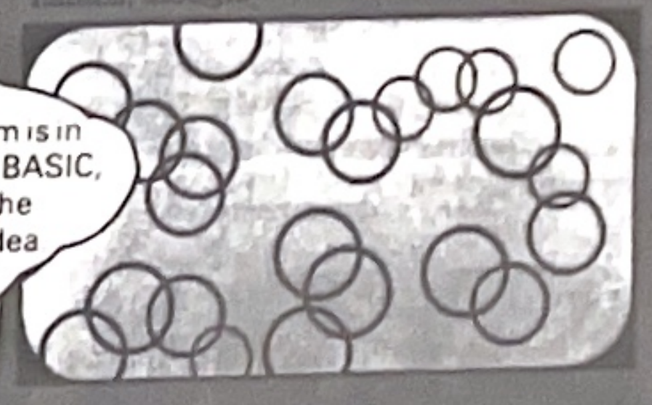
```
100 INPUT "DISTANCE";M
110 INPUT "TIME";T
120 GOSUB 200
130 PRINT "AVERAGE SPEED WAS"
140 PRINT M/T;"MPH AND";K/T;"KPH"
150 STOP
200 REM: SUBROUTINE TO CONVERT MILES
210 LET K=M*1.609
220 RETURN
```

This is a subroutine to convert miles to kilometres. You can often use the same subroutine in lots of different programs. Check that you use the same variable names, though.

Circles program

```
1 Centre of circle = X,Y
2 Radius of circle = R
3 Colour = X
4 Gosub 10
5 Goto 1
10 Rem: Subroutine to draw circles
11 Draw a circle with centre X,Y;
   radius R and colour X.
12 Return
```

This program is in English, not BASIC, to give you the general idea



Quiz program