Basic Flavours

LET

Only the Sinclair Spectrum uses the LET part of the instruction. On other computers this is implied, meaning it can be left off. For example, line 20 can be written as A=A + 1 instead of LET A = A + 1

END

This is not used on the Spectrum. The last line of the program typed in is assumed to be the end of the program

GOTO

Appears on the screen as two words (GO TO) on the Spectrum, although only one key is pressed. Most other computers, with the exception of the BBC will accept the instruction typed as two words

following the order of the line numbers. By the time we got to line 60 the value of A had already been changed to 8, and that is what it will print. Finally we come to:

```
70 END
```

The END statement tells BASIC that the end of the program has been reached. Some versions of BASIC insist that all programs should finish with END while others do not (see the 'Basic Flavours' box).

Notice that when you run the program it only 'works' once. To get it to go through once more you have to type RUN<CR> again. Now we'll look at a way of getting the program to work as many times as we want by using the GOTO statement.

Using GOTO

The same program but with an extra line is given below. If you have switched off the computer to take a break, type it in. Otherwise all you need to do is to type in lines 70 and 80. These are shown in blue in the listing below.

```
10 REM COMPUTERS NEVER MAKE
   MISTAKES<CR>
20 PRINT "TYPE IN A NUMBER"<CR>
30 INPUT A<CR>
40 LET A = A + 1<CR>
50 PRINT "I THINK THE NUMBER YOU TYPED
   WAS  "; <CR>
60 PRINT A<CR>
70 GOTO 20<CR>
80 END<CR>
```

After you have typed it all in and LISTed it, see if you can figure out what will happen before you try to RUN it. Then type RUN<CR> and, as in the first version of the program, you should see:

```
TYPE IN A NUMBER
```

Type in any number (using the numeral keys) and hit RETURN. The computer will add 1 to the number and display it at the end of the message.

```
I THINK THE NUMBER YOU TYPED WAS 8
```

You will see that this is immediately followed by the TYPE IN A NUMBER message again. Entering another number and hitting return again makes the program cycle like this *ad infinitum*. The reason this happens can be found in line 70:

```
70 GOTO 20
```

When BASIC reaches a GOTO statement, instead of continuing to the next line, it GOes TO the line number specified. Here it is directed back to line 20 and the whole program is run all over again. It goes on looping back like this forever. If you want to stop the program from running you'll find there's no way of getting out of the loop. The program just goes on and on waiting for your input.

As you would expect, there are ways of writing the program so that we can get out of it if we want to, and we'll look at one of these in the next instalment of this course. Meanwhile, we still have to stop the program. If your computer has a BREAK key, it can be used to stop the program from running. Typing RUN<CR> will start the program again.

Notice that we still have the END statement at the end of the program. The way we have written this program, with the GOTO 20 statement creating an endless loop, we never do get to the end, but some versions of BASIC insist that we always use an END at the end!

If you can't find a way of stopping the program, try hitting the RESET key. That is almost certain to halt the program. Then try to LIST it again. If you get a list, you will be able to 'edit' the program in the exercises below. If you do not get a list, it means the RESET on your computer destroys the program in memory and you will then have to type the whole thing in again.

Exercises

These questions are carefully graded and are designed to be fun. Working through exercises is one of the best ways of checking that you have understood the material presented and are making genuine progress.

Before starting the exercises, try changing a few of the lines to see the effect on the way the program runs. You can't possibly do the computer any harm even if you make mistakes or hit the wrong keys. To change a line, type in the program and then check the result by LISTing it. The whole program will appear on the screen again. Type the number of the line you want to change followed by the new line. Try this:

```
10 REM COMPUTERS SOMETIMES MAKE
   MISTAKES<CR>
```

then type LIST again. Notice how the first line has been changed. If you want to get rid of everything in the line, just type the line number followed by <CR>. Try:

```
10<CR>
   LIST
```