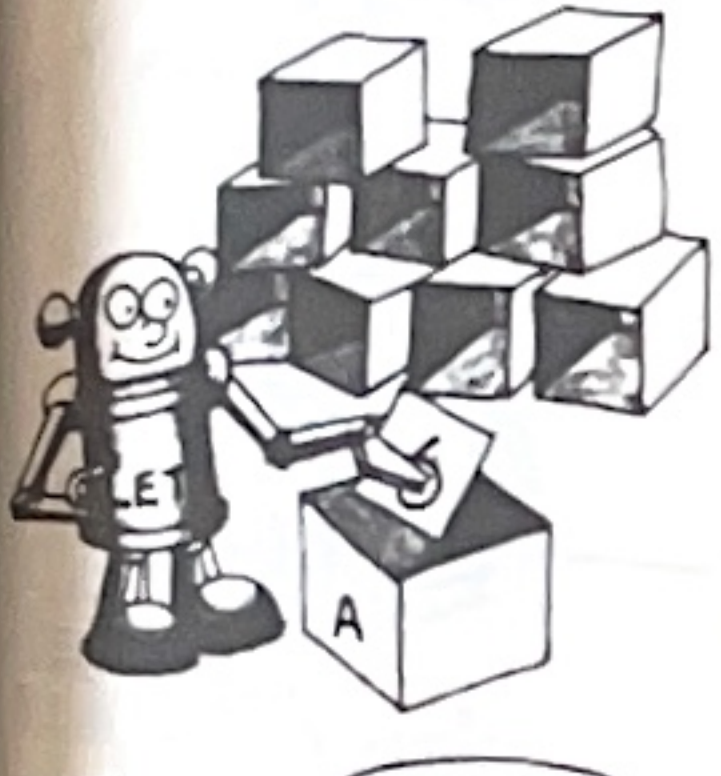
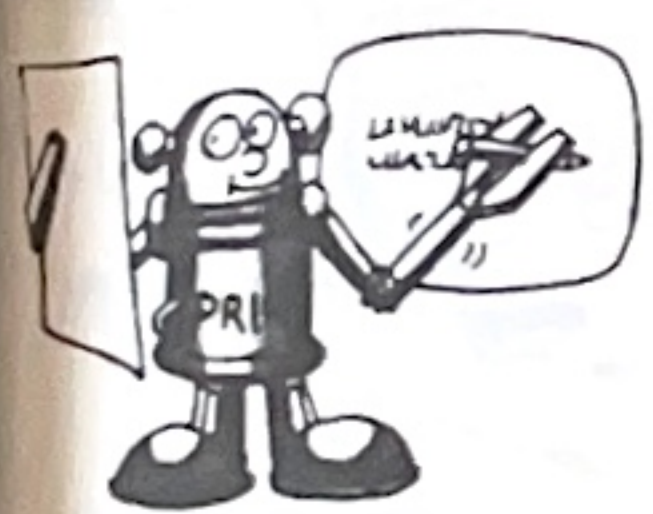
Summary of BASIC

This section lists some common BASIC words and describes what they make the computer do and how they are used. Most of them have been used in the programs in this book, so you can check back through the book to see how they work in a game. Not all the words can be used on all the computers mentioned in this book. The conversion chart on page 46 shows what you can use instead.



LET tells the computer to label a section of its memory and put a particular value in it e.g. LET A=6 means label a section of memory "A" and put the value 6 in it. "A" is called a "variable" and putting something in it is called "assigning a value to a variable".

Some variable labels are followed by a dollar sign e.g. A$. This means they are for "strings", which can contain any number of characters, including letters, numbers and symbols.



PRINT tells the computer to display things on the screen and you can use it in several ways:

A message enclosed in quotation marks with PRINT in front of it will be displayed on the screen exactly as you typed it. The section inside quotes does not have to be in BASIC, it can be anything you like.

PRINT followed by a variable label e.g. PRINT A or PRINT A$ tells the computer to display the contents of that variable on the screen.

PRINT can also do calculations and then display the results e.g. PRINT 6*4 will make the computer display 24.

You can use PRINT by itself to leave an empty line.

RND tells the computer to choose a number at random. Different computers use different forms of RND and you can see what these are in the conversion chart on page 46. On Sinclair computers RND by itself produces a number between 0 and 0.99999999. You can vary the limits of the number it chooses by multiplying RND and adding to it. E.g. RND*20 produces a number between 0 and 19.99999999, while RND*20+1 produces a number between 1 and 20.99999999.

See INT for how to produce only whole numbers.

See CHR$ for how to produce letters and other keyboard characters at random.

INT is short for integer, which means whole number. For positive numbers, it tells the computer to ignore everything to the right of the decimal point. E.g. INT(20.999) is 20. For negative numbers, it ignores everything to the right of the decimal point and "increases" the number to the left of it by one e.g. INT(−3.6) is −4.

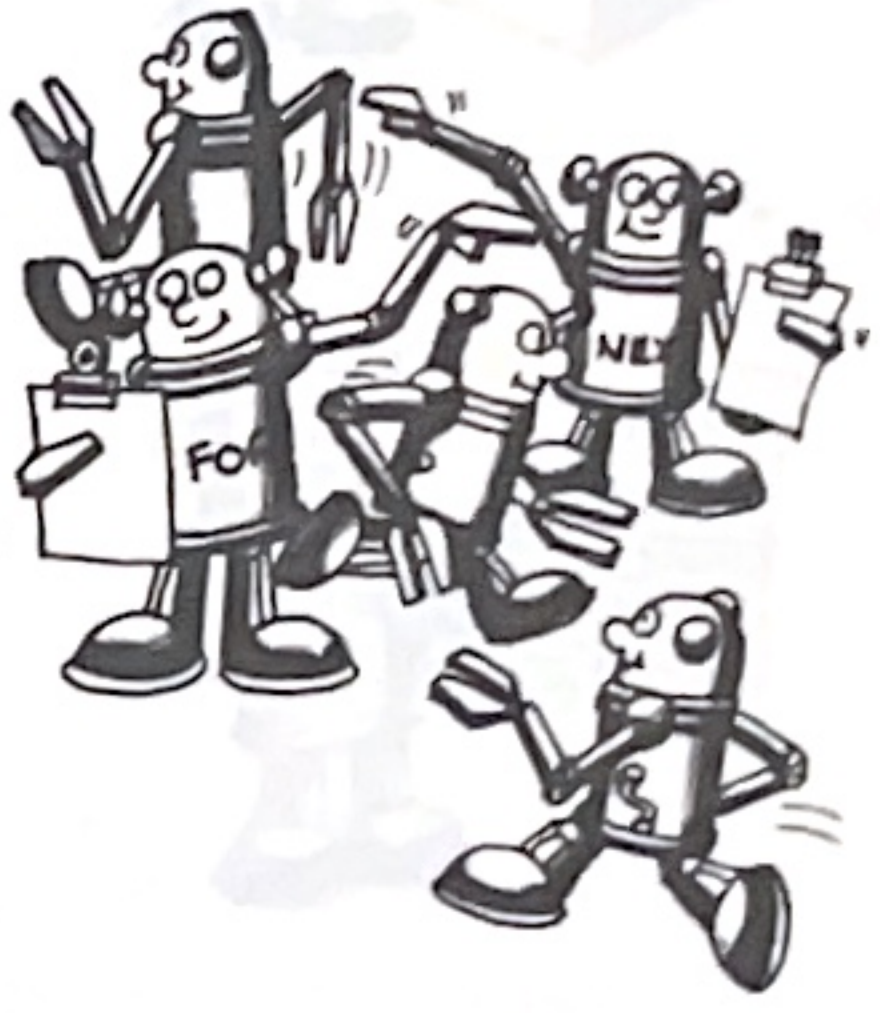
INT is often used with RND, like this: INT(RND*20+1) which tells the computer you want it to choose a whole number between 1 and 20.

CHR$ converts numbers into letters. Apart from the ZX81, all the computers in this book use the ASCII* set of keyboard characters in which each character corresponds to a certain number. E.g. letter A has the code number 65 and PRINT CHR$(65) will display an A on the screen.

You can use CHR$ with INT and RND to make the computer select random letters, like this:

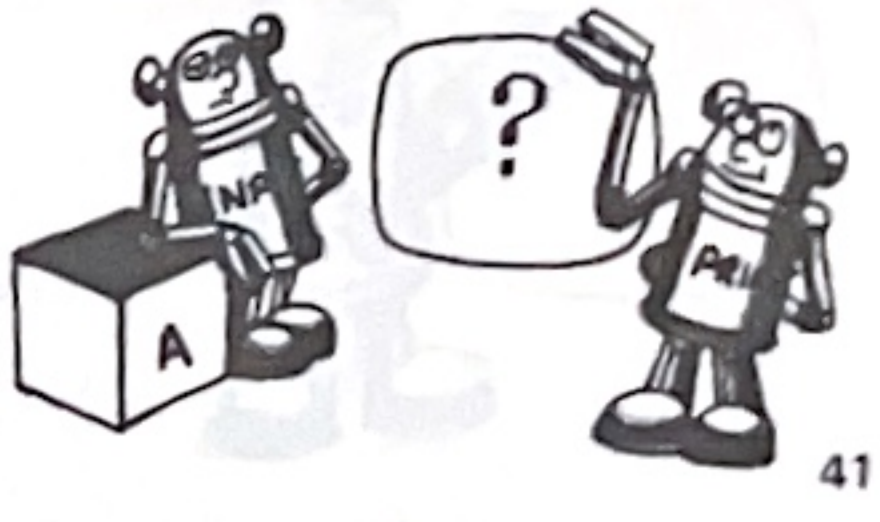
CHR$(INT(RND*26+65))

This line will produce random letters on a ZX Spectrum (see conversion chart for other computers).



FOR is used to start a "loop" which will make the computer repeat part of a program a certain number of times. It must be followed by a variable (such as G to stand for the number of goes allowed in a game), and the variable must be given start and end values (such as 1 TO 10.)

The end of the loop is marked by a NEXT line (NEXT G in this example) which increases the value of the variable by 1 each time and then sends the computer back to the FOR line again. When the variable reaches its end value, the computer ignores the NEXT line and carries on to the line which follows it. Every FOR must have a NEXT or you will get a bug.



INPUT labels a space in the computer's memory, prints a question mark and then waits for you to type something which it can put in this memory space. It will not carry on with the rest of the program until you press RETURN, ENTER or NEWLINE.

You can use number or string variables with INPUT, but if you use a number variable the computer will not accept letters from you.

*American Standard Code for Information Interchange (see page 45)