increments by 1 each time up to the highest temperature) followed directly on the same line (that's why we used the semi-colon) by the words in quotes, followed directly again (another semi-colon) by the value of F. If you look carefully at F, you will see that it is the current value of the Centigrade temperature, converted into Fahrenheit by multiplying it by nine, dividing it by five and then adding 32. The NEXT X line ensures that we go through the conversions until the upper limit in the FOR-NEXT loop has been reached.

Before going on to look at a more sophisticated variation on the PRINT statement, it is worth taking a second look at line 70 in our temperature conversion program:

```
70 LET F = X * 9 / 5 + 32
```

This line assigns a value to the variable F (which stands for Fahrenheit). The program first takes the value of X (the temperature in Centigrade), multiplies it by 9, divides that by 5 and then adds 32. The way this formula would be presented in an ordinary arithmetic book is $F = C \times 9 \div 5 + 32$. BASIC uses * for multiplication, / for division, + for addition and − for subtraction.

In ordinary arithmetic, and in BASIC too, the order in which arithmetical operations are carried out is important. Multiplication always has top priority, followed by division, followed by addition, followed by subtraction. If parts of an arithmetic expression are enclosed in round brackets, they must be evaluated first. If you want an addition to be performed first, before a multiplication, the addition part must be enclosed in brackets. For example, if you wanted to know how much money you had in your current account plus your savings account in dollars, you might express it in part of a program like this:

```
D = (C + S) * 1.5
```

If your current account has £600 (C) and your savings account has £1,300 (S) and there are 1.5 dollars to the pound, you will want to add the pounds first (C + S) and then multiply by 1.5 to convert to dollars. Without the brackets, the value of your savings account would first be multiplied by 1.5 and then the value of your current account would be added to the result — not what you wanted at all! Always be sure to check that the arithmetic parts will be calculated in the right order.

Print Using

In order to look at a final refinement to our temperature conversion program, try typing it in again and RUNing it. Enter, say, −10 as the value for the lowest temperature and 10 as the value for the highest. As we have already seen, the printout on the screen is very ragged. This is because of the semi-colons used in line 80 in order to concatenate (run together) all the parts being printed, instead of printing them on separate lines. Which is fine, except that the space taken up by the figures — both the Centigrade and the Fahrenheit ones — varies. This has the effect of pulling the columns out of alignment and making the printout look untidy.

Almost all versions of BASIC have a special PRINT feature called PRINT USING. It allows the appearance of the printed numbers or words to be 'formatted' or tidied up. If you want to print the value of X and that value is known in advance to range from, say, −99 to 99, the figures can be printed out correctly aligned by using PRINT USING "###";X. The three 'hash' signs allow up to three digits, or two digits preceded by a minus sign, to be printed. If more than three digits are entered, they will not be printed out correctly. If, however, only two digits are entered (or only one) they will be positioned correctly. If decimal points are required, they can be included in the appropriate position within the hash signs. For example, the statement can take the form PRINT USING "###.##";X. Use one 'hash' sign for each digit. All the decimal points will line up automatically.

Modify the original program by changing line 80 and adding lines 82, 84 and 86:

```
80 PRINT USING "###";X;
82 PRINT " IN CENTIGRADE IS ";
84 PRINT USING "###.##";F;
86 PRINT " IN FAHRENHEIT"
```

LIST the program again and then RUN it. All the columns should now be lined up perfectly.

We will find out how to 'save' programs, so that they do not need to be re-typed every time, in the next instalment of the course.

Exercises

- Try entering a 'lowest temperature' of −1000. Why doesn't the program work this time? How would you modify the PRINT USING statement in line 80 to make it work?
- Alter line 84 so that only whole numbers (no decimal fractions) are printed.
- Write a program to convert a range of figures in pounds to dollars, using an exchange rate of $1.50 to the £.

Basic Flavours

PRINT USING

This facility is not available on the Commodore 64, Oric, Spectrum, ZX81 or BBC Micro. However, the BBC can limit the number of decimal places to be printed, and this is achieved by using the following instruction:
@%=131594

COMMA

The use of a comma between print fields will separate the items to be printed by causing the second one to begin at the next mid-line position. The spacing that this gives rise to will vary according to the lengths of the print fields and the number of characters in one line of the particular computer's display