

# Speaking In Tongues

Basic has a familiar mathematical construction, and so is relatively easy to learn, but it is clumsy in relation to some of the other languages

Unless your home computer is a Jupiter Ace (see page 150) then it will almost certainly feature BASIC as its resident programming language. But that doesn't mean to say that you are restricted to that choice, and though BASIC is acknowledged as being a particularly easy language to learn, there are other far more suitable languages for writing specific applications. To install these on your computer it will be necessary either to replace the ROMs containing the BASIC interpreter, or to load the new language into RAM — in which case you will need a machine with a reasonable memory capacity so that there is RAM left over to contain your programs. A few home computers, such as the Sharp MZ-711, have anticipated this problem by also having the BASIC interpreter cassette loaded.

## BASIC-ENGLISH ENGLISH-BASIC

A much-maligned language, BASIC was developed out of FORTRAN (which was one of the first high-level programming languages, and is still the most popular for scientific and engineering applications) as a tutorial introduction to programming, for university students. Because it was intended for self-tutorial use, BASIC is usually interpreted rather than compiled (see page 184), and this factor has been the main reason why it has become the native language of almost all home computers. Interpreted languages are cheap to implement, use comparatively little computer memory, and are extremely suitable for program development.

BASIC is now a powerful language in its own right, but is hindered by its variety of non-standard dialects (every computer's BASIC is unique to that machine), and its lack of specialised data and control structures. These limitations mean that self-taught programmers can develop bad programming technique, and that good programming technique can be difficult to apply to specific problems in BASIC.

This short program illustrates the appeal and generality, but also the limitations, of BASIC:

```
100 INPUT "What's your name?" ;NS
200 INPUT "and how old are you?" ;A
300 FOR K = 1 TO A
400 PRINT K, "Hello" ;NS
500 NEXT K
600 INPUT "Want another go?" ;YS
700 IF LEFTS(YS,1) = "Y" THEN GOTO 100
800 PRINT "Goodbye" ;NS
900 END
```

## PASCAL-ENGLISH ENGLISH-PASCAL

PASCAL was developed in the early 1970's as the successor to BASIC. Its range of data and control structures derive from the FORTRAN/ALGOL group of languages and these are intended to encourage the student to approach programming in a systematic way, and to write well-structured, easily understood code. This is very desirable for developing good programming technique, but it does mean that the early stages of learning programming are harder for the complete beginner — in part because of the discipline that the language imposes, and also because it is usually compiled rather than interpreted. Nevertheless, PASCAL programs tend to be elegant, relatively quickly developed, and much easier to understand.

Here is the PASCAL equivalent of the BASIC program:

```
VAR
NAME      : PACKED
           ARRAY (1..30)
           OF CHAR;
AGE,COUNT : INTEGER;
ANSWER    : PACKED
           ARRAY (1..3)
           OF CHAR;
RUNNING   : BOOLEAN;

BEGIN
  RUNNING := TRUE;
  WHILE RUNNING DO
  BEGIN
    WRITE('What is your name?');
    READLN(NAME);
    WRITE('and how old are you?');
    READLN(AGE);
    FOR COUNT = 1 TO AGE DO
      WRITE(COUNT, 'Hello ');
    WRITE('Want another go?');
    READLN(ANSWER);
    IF ANSWER(1) = 'N'
    THEN RUNNING := FALSE;
  END;
  WRITELN('Goodbye', Name);
END
```

## COMAL-ENGLISH ENGLISH-COMAL

COMAL was developed to combine the accessibility of BASIC with the powerful structures and disciplined approach of PASCAL. It therefore resembles both, and may have been a model for the development of the BBC Micro's BASIC, which has almost developed into a new language. COMAL has been very successful in school computing and in Scandinavia (where it originated), but seems unlikely to displace either of its two forebears as the introductory programming language.

This is the "Hello" Program in COMAL:

```
100 RUNNING := TRUE
200 WHILE RUNNING DO
300   INPUT "What's your name? " ; NS
400   INPUT "and how old are you? " ; A
500   REPEAT
600     FOR K = 1 TO A DO
700       PRINT K, "Hello" ; NS
800     NEXT K
900   INPUT "Want another go? " ; YS
1000  IF YS = "N" THEN RUNNING := FALSE
1100 ENDWHILE
1200 PRINT "Goodbye" ; NS
```