



PLAYING TURTLE

Our LOGO series has so far concentrated on using the language in its 'immediate mode', in which each command is obeyed as it is entered. An alternative mode allows LOGO users to define a sequence of commands as a 'procedure' that is given a specific name. The commands in the procedure will be carried out only when it is called by name.

Procedures may be unfamiliar to many micro programmers, but the concept is found in everyday 'instruction procedures' such as cooking recipes and knitting patterns. Here we'll create a LOGO procedure that we will call SQUARE. (Unlike LOGO commands, which must always be typed in upper case letters, procedure names may be in either upper or lower case.) We begin by typing:

```
EDIT SQUARE
```

The screen will clear, then the message TO SQUARE will appear at the top of the display as a reminder of the procedure name, and EDIT CTRL-C TO DEFINE CTRL-G TO ABORT will be printed at the bottom. This somewhat cryptic message helps to guide your actions while in 'edit' mode. EDIT simply tells you that LOGO is no longer in immediate mode but expects a procedure to be created. Once this is done and you are ready to store your procedure, pressing Control-C will define (record) the listing, while Control-G will allow you to abort the procedure and start again.

You may type whatever you wish in edit mode, but none of the commands will be obeyed at this time. LOGO simply makes a note of your instructions and stores them in its 'dictionary' under the name you have given your procedure. Complete the SQUARE procedure definition by typing the following commands:

```
TO SQUARE
  REPEAT 4 [FD 50 RT 90]
END
```

Now you can tell LOGO to define this procedure (i.e. remember it) by typing Control-C; you will then receive the message SQUARE DEFINED. If you have made a mistake in your definition you can abort the whole process by typing Control-G, in which case you must begin again, or you may use the edit mode's full screen editor to make corrections. This allows you to use the cursor keys to move anywhere within the procedure and insert or delete characters where desired. (The immediate mode's line editor permits corrections to be made only on the last line entered). LOGO has many more commands that make editing long procedures simple; we will consider these later.

Now your procedure has been successfully defined, let's see how it works. Type DRAW to access the graphics screen, then type SQUARE — the commands in the procedure definition will now be obeyed and the turtle will draw the square. As you can see, procedures are used in exactly the same way as the basic LOGO commands — simply type the procedure name and it will carry out the instructions you have defined. The original commands that are present when LOGO is loaded are called 'primitives'. Once you have 'taught' the language a new procedure it can then be used in just the same way as a primitive. In other words, LOGO is an 'extensible' language, so it may be tailored to suit your own particular requirements.

If your procedure does not do what you expected, it is a simple matter to change the definition. As we have defined it, SQUARE draws a square with sides of 50 units in length. Let's assume that you would prefer a smaller square — say, 30 units each side. Return to the editor by typing in:

```
EDIT SQUARE
```

The text making up the procedure is now displayed, and the screen editor can be used to change the 50 to 30. Once this has been done, redefine the procedure with Control-C and type SQUARE to ensure that it is now the size you wanted. As an exercise, try writing procedures to

Abbreviations

EDIT	ED
HIDETURTLE	HT
SHOWTURTLE	ST

PENTAGON

