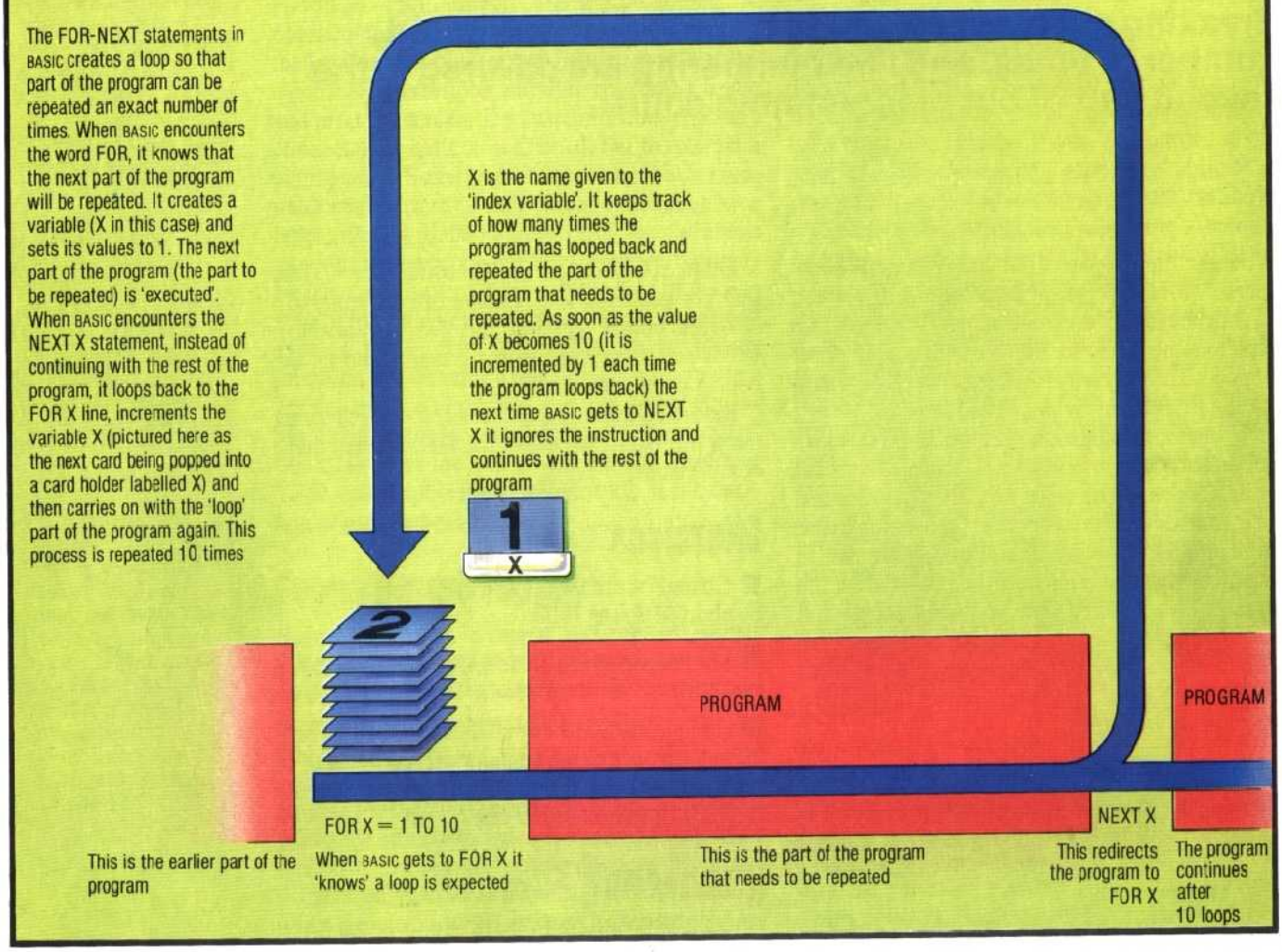
The FOR-NEXT Loop In Basic

The FOR-NEXT statements in BASIC creates a loop so that part of the program can be repeated an exact number of times. When BASIC encounters the word FOR, it knows that the next part of the program will be repeated. It creates a variable (X in this case) and sets its values to 1. The next part of the program (the part to be repeated) is 'executed'. When BASIC encounters the NEXT X statement, instead of continuing with the rest of the program, it loops back to the FOR X line, increments the variable X (pictured here as the next card being popped into a card holder labelled X) and then carries on with the 'loop' part of the program again. This process is repeated 10 times



However, there are occasions, as we learnt in the first part of the course, when using GOTO to make a loop is not the best way of doing things.

Let's return to our old program, modified now to tell the truth this time, to multiply the number input by 10, and do it exactly eight times.

```
10 REM MULTIPLY BY 10
20 FOR X = 1 TO 8
30 PRINT "TYPE IN A NUMBER"
40 INPUT A
50 LET A = A * 10
60 PRINT "YOUR NUMBER MULTIPLIED BY 10
   IS ";
70 PRINT A
80 NEXT X
90 END
```

Type this program in, LIST it to check for mistakes and then RUN it. You will be asked for a number only eight times. After that the program simply stops. The reason this happens is to be found in line 20.

```
20 FOR X = 1 TO 8
```

This is part of a FOR-NEXT loop. It is one of the most useful structures BASIC has to offer. It deserves careful study.

The way we have used it here, we have created a variable called X. (Variables are explained in the first part of the course on page 21.) We could have called it anything (except A — which we are using for something else). FOR must always be used with a corresponding NEXT, but the NEXT will appear later in the program — after the portion to be repeated. The FOR part of a FOR-NEXT loop always has the following form:

```
FOR variable = starting value TO final value
```

In our example FOR X = 1 TO 8 we have called the variable X and given it an initial value of 1. The next part of the program is then executed by the computer; the number we typed in is multiplied by 10 and then printed on the screen. After that we get to NEXT X and the program loops back to where variable X is — in line 20. As soon as it has done that it increments X by 1, so X acquires a value of 2. The part of the program within the FOR-NEXT loop is then executed again. On coming to NEXT again in line 80, the program loops back and increments X to 3.

The program continues to repeat like this until X has been incremented to 8. After that, the loop is terminated; NEXT X does not go back to FOR X and the program continues to the next line.