```
NEW
OK
10 INPUT "THREE POSITIVE NUMBERS" X, Y, Z
20 GOSUB 100  'G.C.D. OF X AND Y RETURNED IN Y
30 LET X = Z
40 GOSUB 100  'G.C.D. OF Z AND G.C.D. RETURNED IN Y
50 PRINT "G.C.D. ="; Y
60 GOTO 10
100 'SUBROUTINE RETURNS G.C.D. OF X AND Y IN Y
110 IF X >= Y THEN 150
120 TEMP = X  'SWITCH TO FORCE X >= Y
130 X = Y
140 Y = TEMP
150 TEMP = X MOD Y
160 IF TEMP <> 0 THEN 130  'KEEP TRYING
170 RETURN  'DONE WHEN X MOD Y = 0
RUN
THREE POSITIVE NUMBERS? 10,35,95
G.C.D. = 5
THREE POSITIVE NUMBERS? 22,121,999
G.C.D. = 1
THREE POSITIVE NUMBERS? ^C
BREAK AT LINE 10
OK
```

The subroutine to compute the G.C.D. of X and Y starts at line 100, and is called from lines 20 and 40. Executing the RETURN of line 170 transfers control to the statement following the GOSUB, i.e. to line 30 or line 50. The remainder operator MOD used in line 150 is explained in Section 3.

READ, DATA and RESTORE

The DATA command lets you insert tables of data into your program, and the READ command gives you access to this data. Execution of a READ command READs the next value from a DATA command and assigns it to a specified variable. Try the following example:

```
NEW
OK
10 READ X
20 PRINT X;
40 DATA 1, 2, 3, 4
RUN
 1
OK
```

If you run out of DATA values, an OD (Out of Data) error occurs. Try adding the following line to the above example.

```
30 GOTO 10
RUN
 1  2  3  4
OD ERROR: 10 READ X

OK
```