subscript of the array X in line 360. It should be examined carefully.

```
360 READ X (I)
```

READ is a new statement we have not encountered before. READ is always used with a corresponding DATA statement. The DATA statement for this line is in line 510:

```
DATA 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 25, 0
```

These numbers, except for the last two, are the numbers of days in each month of the year. The two lines are equivalent to 13 separate LET statements

```
LET X(1) = 31
LET X(2) = 28
LET X(3) = 31
LET X(4) = 30
LET X(5) = 00
LET X(6) = 00
LET X(7) = 31
LET X(8) = 31
LET X(9) = 30
LET X(10) = 31
LET X(11) = 30
LET X(12) = 25
LETX(13)=0
```

The loop set up in line 350 makes I count up from 1 to 13 so we were able to substitute X(I) for X(1), X(2), X(3) etc.

Before returning to this program, let's consider a far simpler small program:

```
10 READ A, B, C
20 LET D = A + B + C
30 PRINT D
40 DATA 5, 10, 20
```

Here, the READ statement in line 10 reads the first item of DATA in line 40 and 'writes' its value into the first variable. In other words, it assigns the value 5 to variable A. READ then reads the next item of data and puts it in the next variable. This program makes A = 5, B = 10 and C = 20. It then adds these and assigns the result to variable D. This result, 35, is then PRINTed in line 30.

Back to the 'Christmas' program. The first time round the loop starting in line 350, the value of I is set out to 1. Line 360 is therefore equivalent to READ X(1). The corresponding data item in line 510 is 31 (the first item). Consequently X(1) is set to 31.

The second time round the loop, I becomes 2 so line 360 is equivalent to READ X(2). The next data item in the DATA line is 28. This means that X(2) is set to 28. In this way all 13 'compartments' in the subscripted variable X are filled up with the number of days in each month; except for the 12th compartment, which has only 25 days in it, and the 13th, which has 0. (Can you see why?)

```
390 GOSUB 750 REM 'LEAP YEAR' ROUTINE
```

This line directs the program to a subroutine that checks if the year entered is a leap year or not.

```
100 REM  LIST OF VARIABLES
110 REM
120 REM  D = TODAY'S DATE
130 REM  M$ = NAME OF MONTH
140 REM  Y = YEAR
150 REM  I = INDEX 1
160 REM  X = ARRAY FOR DAYS IN EACH MONTH
170 REM  R = REMAINING DAYS
180 REM  M = NO. OF MONTH
190 REM  L = INDEX 2
200 REM  Z = INT. VALUE OF Y/4
210 REM
220 REM
230 PRINT "THIS PROGRAM CALCULATES"
240 PRINT "THE NUMBER OF DAYS REMAINING"
250 PRINT "UNTIL CHRISTMAS"
260 PRINT
270 PRINT "ENTER TODAY'S DAY,MONTH,YEAR"
280 PRINT "E.G. 12,JULY,1984"
290 PRINT
300 DIM X(13)
310 INPUT D,M$,Y
320 REM
330 GOSUB 560  REM 'NO OF MONTH' ROUTINE
340 REM
350 FOR I = 1 TO 13
360 READ X(I)
370 NEXT I
380 REM
390 GOSUB 750  REM  'LEAP YEAR' ROUTINE
400 REM
410 LET R = X(M) - D
420 FOR L = M TO 11
430 LET M = M + 1
440 LET R = R + X(M)
450 NEXT L
460 REM
470 IF R = 1 THEN GOTO 500
480 PRINT "THERE ARE";R;"DAYS LEFT UNTIL CHRISTMAS"
490 GOTO 520
500 PRINT "THERE IS 1 DAY LEFT UNTIL CHRISTMAS"
510 DATA 31,28,31,30,31,30,31,31,30,31,30,25,0
520 END
530 REM
540 REM
550 REM
560 IF M$ = "JANUARY" THEN LET M = 1
570 IF M$ = "FEBRUARY" THEN LET M = 2
580 IF M$ = "MARCH" THEN LET M = 3
590 IF M$ = "APRIL" THEN LET M = 4
600 IF M$ = "MAY" THEN LET M = 5
610 IF M$ = "JUNE" THEN LET M = 6
620 IF M$ = "JULY" THEN LET M = 7
630 IF M$ = "AUGUST" THEN LET M = 8
640 IF M$ = "SEPTEMBER" THEN LET M = 9
650 IF M$ = "OCTOBER" THEN LET M = 10
660 IF M$ = "NOVEMBER" THEN LET M = 11
670 IF M$ = "DECEMBER" THEN LET M = 12
680 RETURN
690 REM
700 REM
710 REM
720 REM NOTE: THIS ROUTINE DOES NOT CHECK
730 REM       FOR LEAP YEARS AT THE END OF
740 REM       EACH CENTURY
750 LET Y = Y / 4
760 LET Z = INT(Y)
770 IF Y - Z = 0 THEN GOTO 790
780 RETURN
790 LET X(2) = X(2) + 1
800 RETURN
```