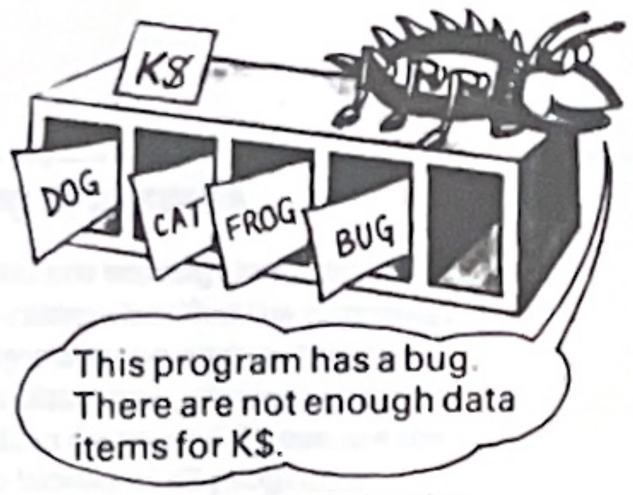
5 Making space for variables

```
5 DIM K$(5)
10 FOR I=1 TO 5
20 READ K$(I)
30 NEXT I
40 STOP

60 DATA DOG, CAT, FROG, BUG
```

This is the size of the variable, i.e. 5 items in a row.

This line puts the data in K$ each time the loop is repeated.



At the beginning of the program you have to tell the computer how big you want the variable to be. You do this with the word DIM followed by the variable name and the number of data items, e.g. DIM K$(5).

For a two dimensional array you give the computer the number of rows and columns in the variable, e.g. DIM C$(5,3). You must always have the right number of data items for the variable or you get a bug.

6 Printing out the data

```
200 LET A=0
210 LET B=0
220 LET A=A+1
230 IF A=6 THEN STOP
240 PRINT A$(A)
250 LET B=B+1
260 LET N=INT(RND(1)*4+1)
270 PRINT B$(B,N)
280 IF B=3 THEN GOTO 250
290 IF B=5 THEN GOTO 250
300 GOTO 220
310 END
```

A keeps count of the number of times this section of the program is repeated.

B keeps count of the data word rows and makes sure that the correct row is used with each data line.

Lines 280 and 290 make the computer print out words from another data word row before printing the next data line.

This sends the computer back to print the next data line.

The computer needs these lines to print out the data lines and words in the right order. This section of the program is repeated five times. Each time, the computer prints out data line number A and some data words from row number B. The actual data words which are chosen are decided by random number N.

Sinclair computers and variables

This program does not work in its present form on Sinclair computers because they handle strings in a different way.

To tell a Sinclair computer to pick out a particular data item from a variable you have to give it the numbers of the first and last characters of the item you want. This is the same system as the Sinclairs use for LEFT$, RIGHT$, etc. (See page 32.)

For two-dimensional arrays you have to tell the computer the number of the row as well as the numbers of the characters. For instance, A$(2, 4 TO 6) is PUG.

At the beginning of a program you tell the computer how many rows the array has, and how many characters there are in each row, e.g. DIM A$(2,9) means two rows, each with nine characters. All the rows in the array must have the same number of characters.

The complete funny poems program

Now you can put the parts of the program together and write the complete poetry program. The first part of the program (lines 10 to 190) give the computer the data and the second part (lines 200 to 310) prints out the poem. Each time you run the program you get a different version of the poem because the random number N makes the computer pick different words.

```
10 DIM A$(5)
20 DIM B$(7,4)
30 FOR I=1 TO 7
40 FOR J=1 TO 4
50 READ B$(I, J)
60 NEXT J
70 NEXT I
80 DATA TASHKENT, TRENT, KENT, GHENT
90 DATA WRAPPED UP, COVERED, PAINTED, FASTENED
100 DATA HEAD, HAND, DOG, FOOT
110 DATA IN A TENT, WITH CEMENT, WITH SOME SCENT, THAT WAS BENT
120 DATA IT RAN OFF, IT GLOWED, IT BLEW UP, IT TURNED BLUE
130 DATA IN THE PARK LIKE A QUARK, FOR A LARK, WITH A BARK
140 DATA WHERE IT WENT, ITS INTENT, WHY IT WENT, WHAT IT MEANT
150 FOR I=1 TO 5
160 READ A$(I)
170 NEXT I
180 DATA THERE WAS A YOUNG MAN FROM, WHO, HIS
190 DATA ONE NIGHT AFTER DARK, AND HE NEVER WORKED OUT
200 LET A=0
210 LET B=0
220 LET A=A+1
230 IF A=6 THEN STOP
240 PRINT A$(A)
250 LET B=B+1
260 LET N=(RND(1)*4+1)
270 PRINT B$(B,N)
280 IF B=3 THEN GOTO 250
290 IF B=5 THEN GOTO 250
300 GOTO 220
310 END
```

Lines 10 and 20 tell the computer how much space to leave for the variables – a row of 5 for A$ and 7 rows of 4 for B$.

These are the nested loops for putting the data in B$.

Lines 80 to 140 contain all the data words to be stored in B$.

This is a loop to put the data into A$.

Lines 180 to 190 contain all the data lines to be stored in A$.

This prints the data line stored in A$ compartment number A.

This prints the data words stored in B$ row B, column N.

The program stops at line 230 when A=6, so it never reaches line 310, but some computers need an END anyway.

Sample runs

```
THERE WAS A YOUNG MAN FROM
KENT
WHO
WRAPPED UP
HIS
HEAD
IN A TENT
ONE NIGHT AFTER DARK
IT GLOWED
LIKE A QUARK
AND HE NEVER WORKED OUT
WHY IT WENT
```

```
THERE WAS A YOUNG MAN FROM
GHENT
WHO
PAINTED
HIS
FOOT
WITH CEMENT
ONE NIGHT AFTER DARK
IT TURNED BLUE
WITH A BARK
AND HE NEVER WORKED OUT
ITS INTENT
```

Here are two of the 16,384 possible different versions of the poem. If you try this program and always get the same poems, look in your manual for how to make the computer produce different random numbers. Some computers produce the same sequence of random numbers each time they are switched on.