

OPEN FORUM

Clay Pigeon Shoot

on Lynx

An addictive game for the Lynx. A clay

pigeon flies across the screen and you shoot it using the space bar. The computer will sound a bleep if you hit it.

The skill levels determine how fast the pigeon flies. You have 20 shots and at the end are awarded your score.

```
100 REM *** CLAY PIGEON SHOOT ***
110 REM *** JOANNA GREENWOOD ***
120 LET J=0
130 LET T=0
140 LET F=125
150 PAPER 0
160 INK 7
170 CLS
180 INPUT "SKILL LEVEL (1 TO 8)";E
190 IF E>8 THEN GOTO 180
200 LET A=5
210 LET B=6
220 CLS
230 PLOT 4,F,240
240 PLOT 0,F-1,241
250 PLOT 2,F+1,241
260 IF A>250 THEN GOTO 200
270 PLOT 4,B,5
280 PLOT 4,A,6
290 PLOT 4,B,6
300 PLOT 4,A,5
310 IF KEY$=" " THEN GOTO 400
320 IF INK=0 THEN GOTO 350
330 INK 0
340 GOTO 230
350 LET A=A+E
360 LET B=B+E
370 IF KEY$=" " THEN GOTO 400
380 INK 7
390 GOTO 230
400 INK 7
410 LET J=J+1
420 PLOT 0,F,240
430 PLOT 2,F,5
440 IF INK=0 THEN GOTO 400
450 INK 0
460 GOTO 420
470 IF A=F THEN GOTO 500
480 IF B=F THEN GOTO 500
490 ELSE GOTO 600
500 LET T=T+1
510 BEEP 50,100,63
520 INK 7
530 PLOT 4,F-3,2
540 PLOT 4,F+3,2
550 PLOT 4,F-3,8
560 PLOT 4,F+3,8
570 PAUSE 2500
580 CLS
590 GOTO 630
600 IF J>19 THEN GOTO 670
610 INK 7
620 GOTO 270
630 INK 7
640 LET A=5
650 LET B=6
660 GOTO 230
670 CLS
680 INK 7
690 PAUSE 5000
700 VDU 24
710 IF T<>1 THEN LET O$="S"
720 ELSE LET O$=" "
730 PRINT "YOU SCORED ";J;" POINT";O$
740 PRINT "USING 20 SHOTS"
750 VDU 25
760 PRINT @ 3,70;"WOULD YOU LIKE ANOTHER GO (Y/N)"
770 IF GET$="Y" THEN GOTO 120
780 END
```

Clay Pigeon Shoot
by Joanna Greenwood

Format

on Dragon

Dragon and other micro users will find this set of routines to be a time saver. They allow a programmer to define sentences and have them printed on the screen without having to count up sentence length to avoid "wrap-around". The routines are particularly useful for displaying game instructions, pages of information, etc.

Lines 10-60 show how sentences are

defined, while lines 1000-1050 and 1060-1090 are the routines which handle the screen formatting.

The subroutine at 1000 adds a space to the sentence or phrase, since a space is used as a cue to print a word. Missing this out will mean that the last word of a sentence is not printed.

L\$ is used to hold each character of the sentence in turn. If L\$ is a space, Chr\$(32), the subroutine at 1060 is called before the next group of characters is processed. W\$ is used to hold each group

of characters.

The subroutine at 1060 calculates whether or not a word will fit on to the current print line. Pos (1) returns the current horizontal print position — note that the routine could be adapted for use with a printer by using Pos (-2). If a word will not fit on to the print lines, a line feed (Chr\$(13)) is printed before the word is displayed.

Removing the Rem statement in line 1080 will cause each new sentence to be printed on a new line.

```
10 NC = 32: REM SCREEN WIDTH (NUMBER OF
    COLUMNS)
20 CLS
30 A$ = "THIS IS THE FIRST SENTENCE WHICH
    IS TOO LONG TO FIT ONTO A SINGLE LINE.
    ":GOSUB1000
40 A$ = "THIS IS THE SECOND SENTENCE,
    WHICH IS ALSO TOO LONG TO BE DISPLAYED
    ON ONE LINE.": GOSUB 1000
50 REM REST OF PROGRAM
60 END
```

```
1000 L$ = " ": W$ = " ": A$ = A$ + CHR$(32)
1010 PRINT STRING$(3,32);
1020 FOR I = 1 TO LEN(A$)
1030 L$ = MID$(A$,I,1): W$ = W$ + L$
1040 IF L$ = CHR$(32) THEN GOSUB 1060
1050 NEXT: RETURN
1060 IF POS(1) + LEN(W$)>NC THEN PRINT CHR$(13);
1070 PRINT W$;
1080 REM IF MID$(W$,LEN(W$)-1,1) = "." THEN PRINT CHR$(13);
1090 W$ = " ": RETURN
```

Format
by B Skinner