2. Life

General Description

LIFE has been about in many forms for several years. I first wrote it in Z80 assembler for the NASCOM four years ago, and I would recommend that any person who understands the logic of LIFE may enjoy doing it in 6502 Assembler.

To the uninitiated, LIFE simulates cell regeneration. Even operating relatively slowly, as it does in BASIC, it produces a generation every seven seconds creating some lovely results including splendid examples of cell division. The rules of cell regeneration are:

a cell DIES if it has one or less immediate neighbours, or more than three neighbours,

a cell is STABLE if it has just two neighbours,

a new cell is BORN if the location at which it will be created has just three neighbours.

The more ghoulish friends of mine have invented 'CANCER' as a derivative of this simulation.

After the pretty multi-coloured introductions, you decide where you will place a cell in the closed environment of your screen and use the copy key to copy it on to the screen. When satisfied with your creation press the SPACE BAR to wait for the next generation to be computed. ESC gets you out of the program when you have found a stable cell structure or when you are bored.

Detailed Description

Lines 10-190 Here is a pleasure of BBC Basic. A reasonably complicated process is reduced in the main structure to relative simplicity. Line 120 must be altered for version 1.0 and above

operating systems - see the manual.

200-390 Very pretty double size characters are drawn all over the screen. Line 270 decides the number of rows, line 280 doubles up the character size and line 290 determines the number of LIFEs on the screen. The RND is used to control the colour of the display

400-650 There is an assembly routine in the manual to read the character on the screen, but this routine simply keeps track of the cursor control keys and calls the toggle procedure to put a cell on or take a cell of the screen. Buffer clearing at line 490 is essential.

660-720 This routine uses the copy key to exchange the position on the screen between a star and no star.

730-840 This routine simply draws the closed environment of the screen.

850-1070 The instructions here are a bit more specific than the rules above. There are actually only three rules for LIFE but for children we have elucidated a bit.

1080-1320 The guts of the operation. Consider a cell pattern as below:

( )	( )	(1)
( )	(*)	( )
(2)	(3)	( )

For any and every position on the screen, if the star represents the point currently being computed each of the other positions marked must be scanned for an alive or non-existent cell. If there are cells at 1, 2 and 3 then the star will be regenerated. This routine does this scan.

1330-1430 This displays the new cell structure.

1440-end Leave out the error report if you like.

Educational Notes

Apart from looking at pretty patterns, the real point of the program is to see if you can create the environment for cell division. To do this students ought to be issued with graph or patterned paper on which their initial pattern can be recorded or decided before entering it to the program. This simulation is very difficult for the less able, and I found that as a simulation it was best restricted to better fifth formers and sixth formers. On the

other hand, lower down in the school, fun was had by less able youngsters doing the following:

They were explained, in some cases several times, the rules of LIFE and then issued with 30 cells. They designed, each of them, an initial pattern, and a competition winner was the person whose structure was kept regenerating the longest. Those who reached a stable structure had to have the generation of stability recorded. Inevitably, the processing of the patterns had to be at low use times for the micro.

Program Listing

```
 10 REM ============================
 20 REM   LIFE
 30 REM ........................
 40 REM
 50 REM ============================
 60 REM   BY C & S.W.  Feb 1983
 70 REM ........................
 80 REM
 90 ON ERROR GOTO 1480
100 MODE 7
110 DIM OLDCELL%(36,25) ,NEWCELL%(36,25)
120 VDU23;8202;0;0;0;
130 PROC_HEADER
140 PROC_INSTRUCT
150 PROC_SET_UP
160 REPEAT
170   PROC_NEW_GENERATION
180   PROC_TIDY
190 UNTIL FALSE
200 REM ============================
210 REM   PROC_HEADER
220 REM ........................
230 REM
240 DEF PROC_HEADER
250 CLS
260 PRINT
270 FOR I%=1 TO 8
280   FOR J%=1 TO 2
290     FOR K%=0 TO 5
300       PRINT TAB(6*K%);CHR$141;CHR$(128+RND(6));"LIFE";
310     NEXT K%
320     PRINT
330   NEXT J%
340   PRINT
350 NEXT I%
360 NOW=TIME
370 REPEAT
380 UNTIL TIME=NOW+300
390 ENDPROC
400 REM ======================
410 REM PROC_SET_UP
420 REM ....................
430 REM
440 DEF PROC_SET_UP
450 GENERATION =0
```

```
  460 PROC_SCREEN
  470 X%=2:X1%=2
  480 Y%=2:Y1%=2
  490 *FX15,0
  500 *FX4,1
  510 KEYIN=GET
  520 REPEAT
  530   PRINT TAB(X%+1,Y%-1);
  540   IF KEYIN<135 OR KEYIN>139 THEN GOTO 630
  550   IF KEYIN=135 THEN PROC_TOGGLE
  560   IF KEYIN=136 THEN X1%=X%-1
  570   IF KEYIN=137 THEN X1%=X%+1
  580   IF KEYIN=138 THEN Y1%=Y%+1
  590   IF KEYIN=139 THEN Y1%=Y%-1
  600   IF X1%<2 OR X1%>35 THEN X1%=X%
  610   IF Y1%<2 OR Y1%>24 THEN Y1%=Y%
  620   X%=X1%:Y%=Y1%
  630   KEYIN=GET
  640 UNTIL KEYIN=32
  650 ENDPROC
  660 REM ======================
  670 REM   PROC_TOGGLE
  680 REM ......................
  690 REM
  700 DEF PROC_TOGGLE
  710 IF OLDCELL%(X%,Y%)=1 THEN OLDCELL%(X%,Y%)=0:NEWCELL%(X
%,Y%)=0: PRINT " "; ELSE OLDCELL%(X%,Y%)=1:NEWCELL%(X%,Y%)=1
: PRINT "*";
  720 ENDPROC
  730 REM ======================
  740 REM   PROC_SCREEN
  750 REM ......................
  760 REM
  770 DEF PROC_SCREEN
  780 CLS
  790 PRINT CHR$30;CHR$147;CHR$60;STRING$(10,CHR$44);CHR$129
;"Generation ";GENERATION;TAB(27,0);CHR$147;STRING$(10,CHR$4
4);CHR$108;
  800 FOR XX%=1 TO 23
  810   PRINT TAB(0,XX%);CHR$147;CHR$53;CHR$130;TAB(37,XX%);
CHR$147;CHR$106;
  820 NEXT
  830 PRINT TAB(0,24);CHR$147;CHR$45;STRING$(36,CHR$44);CHR$
46;
  840 ENDPROC
  850 REM ========================
  860 REM   PROC_INSTRUCT
  870 REM ........................
  880 REM
  890 DEF PROC_INSTRUCT
  900 CLS
  910 FOR X=0 TO 1
  920   PRINT TAB(16,X);CHR$141;CHR$130;"LIFE"
  930 NEXT X
  940 PRINT'" This is the game of life.  The aim  isto set u
p  a  pattern  of cells on  thescreen which stay alive."
  950 PRINT'"  The rules are simple :-"
  960 PRINT"     1. A cell with no neighbours dies."
  970 PRINT"     2. A cell with 1 neighbour dies."
  980 PRINT"     3. A cell with 2 or 3 neighbours"
  990 PRINT"        lives to the next generation."
 1000 PRINT"     4. A cell with 4 or more neigh-"
 1010 PRINT"        bours lives."
 1020 PRINT"     5. IF there are 3 neighbours a new"
 1030 PRINT"        cell is born."
 1040 PRINT'"  Use the cursor keys to move around andthe 'CO
PY' key to insert or delete cells('*').  When  you  have  co
mpleted  yourpattern, press the 'SPACE BAR'"
 1050 PRINT TAB(0,23);CHR$129;"Press any key to continue."
```

```
 1060 Z$=GET$
 1070 ENDPROC
 1080 REM ============================
 1090 REM   PROC_NEW_GENERATION
 1100 REM ........................
 1120 DEF PROC_NEW_GENERATION
 1130 GENERATION=GENERATION+1
 1140 PRINTTAB(24,0);GENERATION;
 1150 FOR Y%=2 TO 24
 1160   PRINT TAB(3,Y%-1);
 1170   FOR X%=2 TO 35
 1180     Z%=0
 1190     IF OLDCELL%(X%-1,Y%-1)=1 Z%=Z%+1
 1200     IF OLDCELL%(X%-1,Y%)=1   Z%=Z%+1
 1210     IF OLDCELL%(X%-1,Y%+1)=1 Z%=Z%+1
 1220     IF OLDCELL%(X%,Y%-1)=1   Z%=Z%+1
 1230     IF OLDCELL%(X%,Y%+1)=1   Z%=Z%+1
 1240     IF OLDCELL%(X%+1,Y%-1)=1 Z%=Z%+1
 1250     IF OLDCELL%(X%+1,Y%)=1   Z%=Z%+1
 1260     IF OLDCELL%(X%+1,Y%+1)=1 Z%=Z%+1
 1270     IF OLDCELL%(X%,Y%)=1 AND Z%<>2 AND Z%<>3 THEN NEWC
ELL%(X%,Y%)=0
 1280     IF OLDCELL%(X%,Y%)=0 AND Z%=3 THEN NEWCELL%(X%,Y%)
=1
 1290     IF NEWCELL%(X%,Y%)=0 THEN PRINT" "; ELSE PRINT"*";
 1300   NEXT
 1310 NEXT
 1320 ENDPROC
 1330 REM ======================
 1340 REM  PROC_TIDY
 1350 REM ......................
 1360 REM
 1370 DEF PROC_TIDY
 1380 FOR X%=2 TO 35
 1390   FOR Y%=2 TO 24
 1400     OLDCELL%(X%,Y%)=NEWCELL%(X%,Y%)
 1410   NEXT
 1420 NEXT
 1430 ENDPROC
 1440 REM ========================
 1450 REM here we have the errors!!
 1460 REM ......................
 1470 REM
 1480 IF ERR=17 THEN GOTO 1530
 1490 MODE 7
 1500 REPORT
 1510 PRINT " at line ";ERL
 1520 END
 1530 *FX4,0
 1540 *FX12,0
 1550 *FX15,0
 1560 MODE 7
 1570 END
>
```