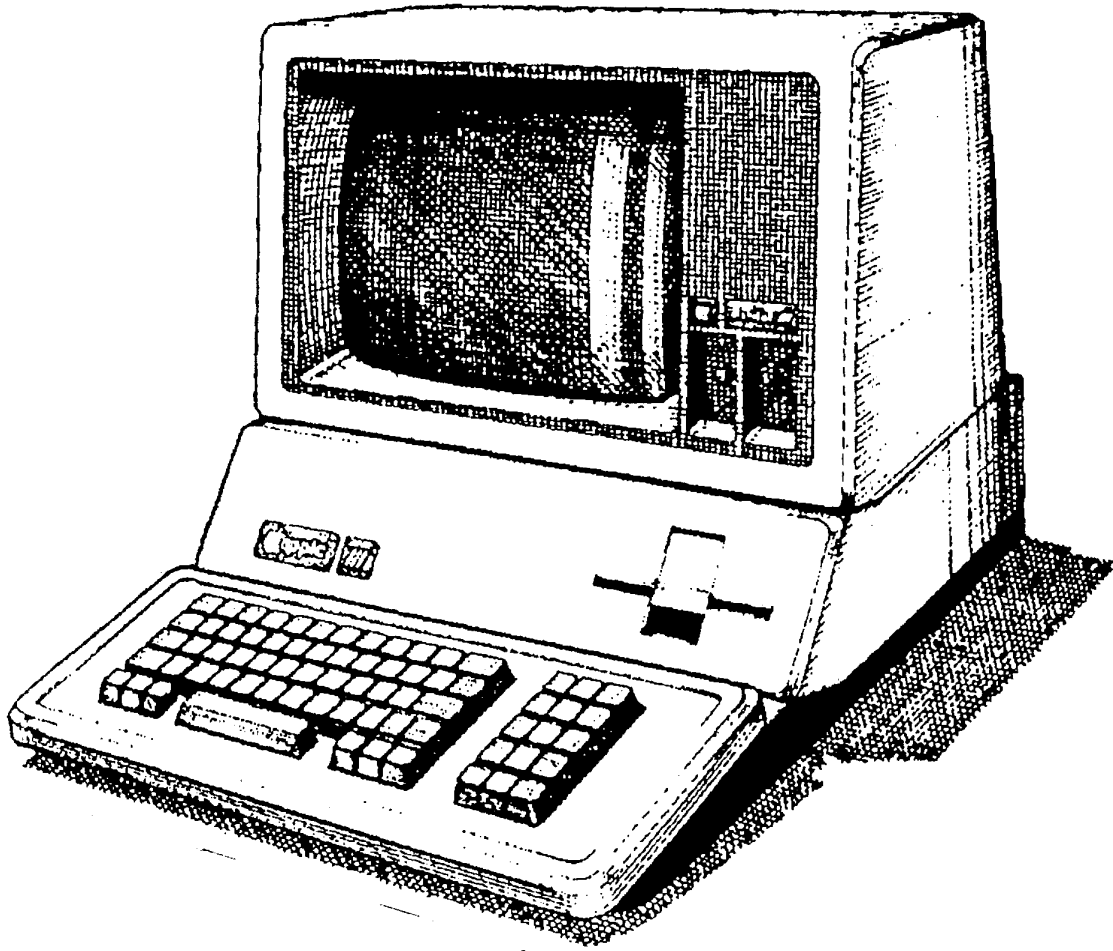


SEE DOC #128



Apple /// Computer Information



DOCUMENT NAME	#
<i>GAME OF LIFE IN BUSINESS BASIC</i>	108

Ex Libris David T. Craig

/// Forever !!!

Apple III Computer Information

Compiled by
David T. Craig

4 pages

736 Edgewater, Wichita, Kansas 67230

Section: Apple III [13]	Contributer: 71076,1173	Size: 7552
Submitted: 4/14/85	File Type: Binary	Access Count: 80
Title: (Conway's game of Life in Basic)		
Keys: GRAPHICS GAME JEPPSON BASIC		

Conway's game as adapted by C. Hood. Written in Basic. There was an associated article in SOFTALK by Jeppson. The program was obtained by permission and is release to MAUG by permission of the SOURCE Apple III SIG to preserve the Apple III data base - E. Raba

BUSINESS BASIC

```

10 REM CONWAY'S LIFE GAME. From an article in the June '82 issue of Softalk
15 REM Magazine by Dr. John Jeppson entitled "Counting With Colors on the
20 REM Apple III" (p.170). Used with permission from Softalk Magazine.
25 REM Program adapted for The Apple III Users Group by Charlie, TCK889.
30 REM Members are urged to read the article in order to get the full back-
35 REM ground on this BUSINESS BASIC program. (Check pathname, line #240)
40 ::
45 REM -- MAIN PROGRAM -- (8/23/82)
50 GOSUB 200:REM initialize
60 GOSUB 400:REM first generation
70 GOSUB 600:REM fix colors
80 ON KBD GOTO 120
90 count=count+1:GOSUB 1000:REM (begin loop) calculate next generation
100 GOSUB 1200:GOTO 90:REM show next generation
110 ::
120 IF KBD=27 THEN 170
130 IF KBD=82 OR KBD=114 THEN PERFORM initgrafix:TEXT:HOME:GOSUB 270:GOTO 60
140 IF KBD=72 OR KBD=104 THEN PERFORM initgrafix:TEXT:HOME:GOSUB 260:GOTO 60
150 GET Y$:ON KBD GOTO 120
160 RETURN
170 FOR i=1 TO 3:PERFORM release:NEXT:TEXT:HOME:PRINT"Quit":END
180 ::
190 REM -- (start) initialize --
200 leftedge=2:REM constants chosen for max number of whole cells
210 rightedge=136:REM each cell is 2X4 dots
220 bottomedge=5:REM y-coord. are odd because they are "top" of
230 topedge=189:REM 2-dot counter 0,1; 2,3; etc.
240 INVOKE".profile/inv/bgraf.inv":REM use appropriate pathname !!!
250 DIM pic$(12):REM the $ is essential; it indicates integer array
260 GOSUB 2000:REM display instructions, menu, and generation count
270 GOSUB 2200:REM make image
275 count=0
280 oldleft=leftedge
290 oldright=rightedge
    
```

/// Forever !!!

```

300 oldbottom=bottomedge
310 oldtop=topedge
320 PERFORM grafixmode(%3,%1):REM color 140X192, buffer 1
330 PERFORM fillport
340 PERFORM grafixon
350 RETURN
360 ::
370 REM -- (start) first generation
400 GOSUB 2400:REM reset window
405 PERFORM moveto(%newleft,%newtop):REM starting position
410 GOSUB 500:REM get pattern
415 GOSUB 2600:REM update window
420 RETURN
499 REM -- subroutine getpattern
500 x= EXFN%.xloc:y= EXFN%.yloc
505 PERFORM dotat(%x,%y+1):REM cursor "floats" just above counter
510 GET g$
515 PERFORM pencolor(%):REM black
520 PERFORM dotat(%x,%y+1):REM remove cursor
525 PERFORM pencolor(%15):REM white
530 PERFORM moveto(%x,%y)
535 IF ASC(g$)=27 THEN 170
540 IF ASC(g$)=8 THEN IF x>leftedge THEN PERFORM moverel(%-2,%0)
545 IF ASC(g$)=10 THEN IF y>bottomedge THEN PERFORM moverel(%0,%-4)
550 IF ASC(g$)=11 THEN IF y<topedge THEN PERFORM moverel(%0,%4)
555 IF ASC(g$)=21 THEN IF x<rightedge THEN PERFORM moverel(%2,%0)
560 IF g$=" " THEN PERFORM drawimage(@pic%(0),%2,%0,%2,%2,%2):REM zap it
565 IF g$="x" OR g$="X" THEN PERFORM drawimage(@pic%(0),%2,%0,%0,%2,%2):GOSU
B 2800:REM plot counter and expand window
IF g$="a" OR g$="A" THEN RETURN:ELSE GOTO 500
570
598 ::
599 REM -- (start) fix colors
600 FOR col=0 TO 15:PERFORM setctab(%0,%col,%col):NEXT col:REM neuter black
610 FOR col=0 TO 3:PERFORM setctab(%9,%col,%(col+1)):NEXT col:REM 'count-up'
620 FOR col=12 TO 15:PERFORM setctab(%9,%col,%(col-1)):NEXT col:REM 'down'
630 PERFORM setctab(%9,%4,%4):REM stop counting up
640 PERFORM setctab(%9,%11,%11):REM stop counting down
650 FOR col=0 TO 15
660 PERFORM setctab(%15,%col,%0):REM white clears screen -- except
670 IF col=3 OR col=12 OR col=13 THEN PERFORM setctab(%15,%col,%15)
690 NEXT col:REM These have right no. of 'influences' for next generation
700 RETURN
998 ::
999 REM -- (start) calculate next generation
1000 GOSUB 2400:REM reset window
1010 PERFORM pencolor(%9):REM orange, an influential color
1015 ON ERR IF ERR=12 THEN RUN:ELSE PRINT"ERROR " ERR", LINE " ERR:LIN:END
1017 REM Handles stack error if <H> or <R> used too many consecutive times
1020 FOR x=oldleft TO oldright STEP 2
1030 FOR y=oldbottom TO oldtop STEP 4
1040 PERFORM moveto(%x,%y)
1050 IF EXFN%.xycolor>10 THEN GOSUB 2800:PERFORM moverel(%-2,%4):PERFORM
drawimage(@pic%(0),%2,%0,%0,%6,%12)
1060 NEXT y,x:OFF ERR
1070 REM (Above) Expand window, plot stamp
1080 GOSUB 2600:REM update window
1090 RETURN
1198 ::
1199 REM -- (start) show next generation --
1200 PERFORM viewport(%oldleft,%oldright,%oldbottom-1,%oldtop)

```

/// Forever !!!

```

1210 REM oldbottom is the "top" dot of two-dot center
1220 PERFORM fillcolor(%15):REM white
1230 PERFORM fillport
1240 PERFORM fillcolor(%0):REM black
1250 PERFORM viewport(%0,%200,%0,%200)
1260 RETURN
1998 ::
1999 REM -- (start) instruction menu --
2000 TEXT:HOME:UPOS=3
2010 PRINT"CONWAY'S LIFE GAME" (Last count="count")
2015 REM 'Last count' is count of generations in prior run when <H> pressed
2020 PRINT"*****":PRINT
2030 UPOS=6:PRINT"Use arrow keys and 'X' to draw pattern when dot appears:"
2040 PRINT
2050 PRINT"< Arrow > moves special cursor dot in any direction
2060 PRINT"< X > deposits a counter to mark pattern
2070 PRINT"< SPC > clears a counter"
2080 PRINT"< A > accepts pattern and starts run
2090 PRINT"< ESC > quits program":PRINT
2100 PRINT"While the LIFE program is running, you can hit -":PRINT
2110 PRINT"< Any Key > to halt or restart display"
2120 PRINT"< ESC > to quit
2130 PRINT"< R > to Repeat program - start a new pattern WITHOUT this menu"
2140 PRINT"< H > to review this HELP menu before starting new pattern"
2150 PRINT:PRINT:PRINT" -- any key to continue -- < ESC > to quit"
2160 GET g$:IF g%=CHR$(27) THEN 170
2170 RETURN
2198 ::
2199 REM -- (start) make image --
2200 REM
2205 pic%(0)=TEN("A800")
2210 pic%(1)=TEN("A800"):REM see page 314, Business Basic Manual
2215 pic%(2)=TEN("0000"):REM "Creating and Storing a Bit Array"
2220 pic%(3)=TEN("0000")
2225 pic%(4)=TEN("8800"):REM creates an open square "image" thus:
2230 pic%(5)=TEN("8800")
2235 pic%(6)=TEN("0000"):REM * * *
2240 pic%(7)=TEN("0000"):REM * *
2245 pic%(8)=TEN("A800"):REM * * *
2250 pic%(9)=TEN("A800")
2255 pic%(10)=TEN("0000"):REM "corner of image (one counter also used)
2260 pic%(11)=TEN("0000"):REM when placing initial pattern
2265 RETURN
2398 ::
2399 REM -- (start) reset window --
2400 newleft=CONV%((oldleft+oldright)/2):REM begin "new window" ctr.of old
2410 REM "average" position may fall "between" proper cell positions
2420 newright=newleft+2
2430 newbottom=CONV%((oldbotto+oldtop)/2)
2440 IF CONV%(newbottom-1) MOD 4<>0 THEN newbottom=newbottom-2
2450 newtop=newbottom+4
2460 RETURN
2598 ::
2599 REM -- (start) update window
2600 oldleft=newleft-2
2605 IF oldleft<leftedge THEN oldleft=leftedge
2610 oldright=newright+2
2615 IF oldright>rightedge THEN oldright=rightedge
2620 oldbottom=newbottom-4
2625 IF oldbottom<bottomedge THEN oldbottom=bottomedge

```

/// Forever !!!

```
2630 oldtop=newtop+4
2635 IF oldtop>topedge THEN oldtop=topedge
2640 RETURN
2799 ::
2799 REM -- (start) expand window --
2800 IF x<newleft THEN newleft=x
2805 IF x>newright THEN newright=x
2810 IF y<newbottom THEN newbottom=y
2815 IF y>newtop THEN newtop=y
2820 RETURN
```

End of Document