

Victoria  
Sept 86

IT'S SUMMERTIME AND THE SUN IS SHINNING, SO WHAT AM I DOING INDOORS?  
PREPARING YOUR NEWSLETTER OF COURSE.

TONY BIGRAS'S OLD TI 99/4A IS ONCE AGAIN FOR SALE. IT INCLUDES DISK  
DRIVE, F-BOX, 32K MEMORY, RS 232, ETC., ETC., FOR APPROXIMATELY 500  
CANADIAN PLAY DOLLARS. I'VE MISPLACED THE PHONE NUMBER TO CALL; CONTACT  
TONY FOR FURTHER DETAILS AND THE OWNER'S NUMBER.

TALKING ABOUT TONY; HE HAS COME THROUGH AGAIN. THIS MONTH'S FEATURE  
PROGRAM IS A RECURSIVE MAZE SOLUTION DEMO THAT ALLOWS THE USER TO INPUT A  
MAZE CONFIGURATION TO BE SOLVED. TONY'S NOTES SAY: "THIS LITTLE PROGRAM  
MAKES USE OF A SINGLE SUBROUTINE ON LINE 2500 TO SEARCH A MAZE (MADE BY  
PRECEDING LINES). THE SUBROUTINE CALLS ITSELF RECURSIVELY TO SOLVE THE  
MAZE. THE MAZE IS MADE WITH THE [ e, s, d, x, p, ] KEYS. THE "P" LIFTS AND  
DROPS THE PEN. THE "e" STARTS THE MOUSE ON ITS SEARCH".

THE SECOND PROGRAM IS A COMPUTER VERSION OF THE POPULAR GAME  
"MASTERMIND". MY WIFE AND I HAVE INVESTED MANY HOURS IN THIS CHALLENGE AND  
RECOMMEND IT TO INCREASE YOUR POWERS OF DEDUCTIVE REASONING. THIS SECOND  
PROGRAM COMES FROM THE NEWSLETTER OF THE TI RIVERSIDE USERS GROUP IN  
CALIFORNIA.

HAPPY PROGRAMMING, KEN ARMSTRONG

---

## TK-WRITER REVISION

---

From the Jackson County 99'ers, comes this tip on modifying Tom  
Knight's Extended Basic load program for TI-Writer.

Apparently, when going from the Editor to the Formatter, the LOAD  
program reloads the assembly program, not checking to see if it is  
still in memory. The resulting wait can be avoided by making the  
following modifications to the LOAD program. In line 108 where it  
says "DSK1.TKWRITER", put in the program name you are using for  
TK-WRITER.

```
100 CALL CLEAR ;; CALL INIT ;; CALL PEEK(-2043,A,B) ;; IF A<>84 OR  
    B<>75 THEN 108  
102 CALL LOAD(16360,85,84,73,76,73,84,250,212,70,79,82,77,65,84,250,  
    132,69,68,73,84,79,82,250,22)  
104 CALL LOAD(8196,63,232) ;; GOTO 110  
108 CALL LOAD("DSK1.TKWRITER")
```

```

100 REM  JUNE 20.1986
110 REM  RECURSIVE MAZE SOLUTION DEMO
120 REM  TONY BIGRAS
130 REM  644 BELTON AVE.
140 REM  VICTORIA B.C.
150 GOSUB 1000
160 GOSUB 2000
170 GOSUB 3000
180 GOTO 180
999 REM  SET UP SCREEN AND HAVE USER INPUT MAZE CONFIGURATION TO BE SOLVED.
1000 CALL CLEAR
1010 CALL CHAR(33,"FFFFFFFFFFFFFFFF")
1020 CALL HCHAR(1,1,33,32)
1030 CALL HCHAR(24,1,33,32)
1040 CALL VCHAR(2,1,33,22)
1050 CALL VCHAR(2,32,33,22)
1060 ROW=12
1070 COL=16
1080 CALL HCHAR(1,16,42,2)
1095 OLDCHR=32
1100 CALL KEY(0,K,S)
1104 CALL GCHAR(ROW,COL,OLDCHR)
1105 CALL VCHAR(ROW,COL,33)
1106 CALL VCHAR(ROW,COL,32)
1107 CALL VCHAR(ROW,COL,OLDCHR)
1108 IF PENDOWN=0 THEN 1120
1109 CALL VCHAR(ROW,COL,33)
1120 IF (K<>69)*(K<>83)*(K<>68)*(K<>88)*(K<>80)*(K<>81) THEN 1100
1130 IF K=69 THEN 1150
1140 GOTO 1160
1150 ROW=ROW+-1
1160 IF K=83 THEN 1180
1170 GOTO 1190
1180 COL=COL-1
1190 IF K=68 THEN 1210
1200 GOTO 1220
1210 COL=COL+1
1220 IF K=88 THEN 1240
1230 GOTO 1250
1240 ROW=ROW+1
1250 IF K=81 THEN 1500
1260 IF (ROW=1)+(ROW=24) THEN 1290
1270 IF (COL=1)+(COL=32) THEN 1310
1280 GOTO 1320
1290 ROW=(24-ROW)+ABS(2*(ROW=24))
1300 GOTO 1320
1310 COL=(32-COL)+ABS(2*(COL=32))
1320 IF K=80 THEN 1340
1330 GOTO 1350
1340 PENDOWN=ABS(PENDOWN-1)
1350 CALL SOUND(-100,500,1)
1490 GOTO 1100
1500 RETURN
1995 REM  RECURSIVE MAZE SOLUTION
1996 REM  IN THIS VERSION THE MOUSE SHOWS ITS THINKING ON THE SCREEN
1997 REM  ANOTHER VERSION COULD USE A STACK FOR STORING THE THOUGHT STREAM
2000 FOUNDEXIT=0
2005 NOTE$="N"

```

```
2006 CALL VCHAR(ROW,COL,32)
2010 GOSUB 2500
2020 RETURN
2500 REM THIS ROUTINE IS CALL RECURSIVELY
2501 IF FOUNDEXIT=1 THEN 2900
2502 CALL GCHAR(ROW,COL,OLDCHR)
2503 IF OLDCHR<>42 THEN 2506
2504 FOUNDEXIT=1
2505 GOTO 2900
2506 IF OLDCHR=32 THEN 2510
2507 ROW=ROW+1
2508 GOTO 2525
2510 CALL VCHAR(ROW,COL,ASC(NOTE$))
2525 NOTE$="N"
2530 IF OLDCHR<>32 THEN 2550
2535 ROW=ROW-1
2540 GOSUB 2500
2550 REM CANT GO NORTH
2555 NOTE$="W"
2560 CALL GCHAR(ROW,COL-1,OLDCHR)
2570 IF OLDCHR<>32 THEN 2590
2575 COL=COL-1
2580 GOSUB 2500
2590 REM CANT GO WEST
2595 NOTE$="E"
2600 CALL GCHAR(ROW,COL+1,OLDCHR)
2610 IF OLDCHR<>32 THEN 2630
2615 COL=COL+1
2620 GOSUB 2500
2630 REM CANT GO EAST
2635 NOTE$="S"
2640 CALL GCHAR(ROW+1,COL,OLDCHR)
2650 IF OLDCHR<>32 THEN 2670
2655 ROW=ROW+1
2660 GOSUB 2500
2670 REM CANT GO SOUTH
2680 REM THIS COORIDNATE IS A DEAD END
2690 REM FIND OUT RETURN DIRECTION FROM THIS COORIDNATE
2700 CALL GCHAR(ROW,COL,OLDCHR)
2710 CALL VCHAR(ROW,COL,46)
2720 IF CHR$(OLDCHR)<>"S" THEN 2740
2730 ROW=ROW-1
2735 GOTO 2810
2740 IF CHR$(OLDCHR)<>"N" THEN 2760
2750 ROW=ROW+1
2755 GOTO 2810
2760 IF CHR$(OLDCHR)<>"W" THEN 2780
2770 COL=COL+1
2775 GOTO 2810
2780 IF CHR$(OLDCHR)<>"E" THEN 2796
2790 COL=COL-1
2795 GOTO 2810
2796 IF FOUNDEXIT=1 THEN 2800
2797 PRINT "NO EXIT"
2798 GOTO 180
2800 PRINT "FOUND EXIT"
2801 GOTO 180
2810 REM
2900 RETURN
3000 RETURN
```

