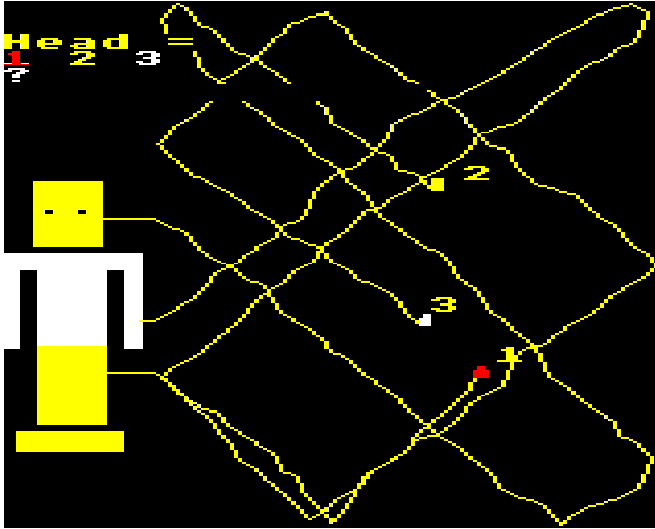


WIRE MAZE



Well at last you have your own robot to cut the grass, clean the car, wash the windows and take the dog for a walk. There is one snag though.

Your robot has been wired up incorrectly. It must have been Friday afternoon when the other robots put your model together. At the moment, if you press the arm-control button, the robot's legs move. You, I'm afraid, are going to have to rewire your new family friend.

How to play

As the program begins, your robot will be drawn on the screen. When the robot shape is completed, the screen will go black and the wiring will be added along with the

control buttons. The robot, complete with wires and buttons will reappear and you will then have to trace the wiring.

Control buttons are:

Red	1
Yellow	2
White	3

At the top of the screen will appear the word "Head" and the three colour buttons. You must decide which of these buttons is connected to the head and press the corresponding number key and RETURN. One wrong try means you must try again, and a subsequent wrong guess will cause your computer to give you the correct answer. If you think about it, if you have guessed two wrong from three, you should know the answer by then anyway!

You continue for the arms and legs.

To play again, with a different maze, re-run the program.

Programming hints

The useful BLOCK procedure is used which draws a rectangle of a specified size at a specified place. The rectangle is created by drawing TWO right-angled triangles joined together. Each triangle is drawn by MOVEing to a point then drawing a line from this point to a third point. If PLOT 85 is used to draw this line from the second point then a filled in triangle is created. You could use this procedure in any of your own non-commercial programs. Copy in the procedure DEF PROC_BLOCK (X,Y,W,H) and call it with PROC_BLOCK (followed by the X and Y position of its bottom left hand corner followed

by its width and height all separated by commas.

The other thing about the program that you might find useful, is the way it draws the maze without you seeing it. This is done by using the VDU 19 command to change all red (1), yellow (2) and green (3) pictures to black (0). The maze is then drawn. Then another set of VDU 19 commands changes them all back to their original colours. The colour numbers are not the same as the second number is the logical colour number; the colour numbering system that is applicable to all modes.

You might find the wire maze too easy. It can be made more difficult by increasing the length of each wire by increasing the larger number in line 420.

```

10  REM  WIRE  MAZE
20  REM  COPYRIGHT  (C)  G.LUDINSKI  1983
30  DIM  XA(3)
40  DIM  YA(3)
50  DIM  W(3)
60  * KEY 10  "OLD|M"
70  MODE  5
80  CLS
90  REM
100 REM  Draw  Robot
110 REM
120 GCOL0,2
130 PROC_BLOCK(70,550,125,125)
140 GCOL0,3:PROC_BLOCK(0,500,275,35)
150 PROC_BLOCK(0,350,35,150)
160 PROC_BLOCK(240,350,39,150)
170 PROC_BLOCK(75,350,125,150)
180 GCOL0,2:PROC_BLOCK(75,200,125,150)
190 PROC_BLOCK(35,150,200,35)
200 COLOUR0:COLOUR130:PRINTTAB(1,12) ".
. ";:COLOUR2:COLOUR128
210 REM
220 REM  Decide  which  controls  connect
to which parts
230 REM
240 FOR I=1 TO 3
250   W(I)=INT(RND(1)*3+1)
260   IF (I=2 AND W(I)=W(1)) OR (I=3 AND
ND (W(I)=W(2) OR W(I)=W(1))) THEN 250
270 NEXT I
280 REM
290 REM  Draw  Wires
300 REM
310 GCOL0,2
320 XA(1)=300:YA(1)=600:MOVE200,600:DR
AW XA(1),YA(1)
330 XA(2)=300:YA(2)=400:MOVE275,400:DR
AW XA(2),YA(2)
340 XA(3)=300:YA(3)=300:MOVE200,300:DR
AW XA(3),YA(3)
350 VDU30

```

```

360 VDU 19,1,0,0,0,0
370 VDU 19,2,0,0,0,0
380 VDU 19,3,0,0,0,0
390 FORI=1TO3
400 XS=(-1)^I:YS=(-1)^I
410 MOVE XA(I),YA(I)
420 FORJ=1TO100
430 DX=XS*INT(RND(1)*30+10)
440 IF ((XA(I)+DX) <300 OR (XA(I)+
DX) >1278) THEN XS=-XS:GOTO430
450 XA(I)=XA(I)+DX
460 DY=YS*INT(RND(1)*20+10)
470 IF ((YA(I)+DY) <0 OR (YA(I)+DY
) >1020) THEN YS=-YS:GOTO460
480 YA(I)=YA(I)+DY
490 DRAW XA(I),YA(I)
500 NEXTJ
510 REM Draw Buttons
520 REM
530 REM
540 GCOL0,W(I):PROC_BLOCK(XA(I),YA(I
),20,10):PROC_BLOCK(XA(I)+5,YA(I)+10,10,
10):GCOL0,2
550 PRINTTAB(INT(XA(I)/64)+1,INT((10
23-YA(I))/32)-1);W(I);
560 NEXTI
570 REM
580 REM Turn Display Back On
590 REM
600 VDU 19,1,1,0,0,0
610 VDU 19,2,3,0,0,0
620 VDU 19,3,7,0,0,0
630 REM
640 REM Write Questions
650 REM
660 FORI=1TO3:PROC_QUESTION:NEXTI
670 END
680 DEFPROC_BLOCK(X,Y,W,H)
690 MOVE X,Y:MOVE X+W,Y
700 PLOT 85,X,Y+H
710 PLOT 85,X+W,Y+H
720 ENDPROC
730 DEFPROC_QUESTION
740 IF I=1 THEN PT$="Head"
750 IF I=2 THEN PT$="Arm"
760 IF I=3 THEN PT$="Leg"
770 PRINTTAB(0,2)PT$;"=":COLOUR1:PRIN
T"1 ";:COLOUR2:PRINT"2 ";:COLOUR3:PRINT"
3 "
780 PRINTTAB(0,4)" ":PRINT"
":VDU11,11
790 T=1
800 PRINTTAB(0,4);:INPUT AN$:IF AN$<>"
1" AND AN$<>"2" AND AN$<>"3" THEN PRINT"
No ";:VDU11:GOTO800
810 IF VAL(AN$)=W(I) THEN PRINT"Yes
":GOTO840
820 PRINT"No, try again":IF T=1 THEN V
DU11,11:T=2:GOTO800
830 IF T=2 THEN VDU11:PRINT"Answer = "
;W(I); " ":A$=INKEY$(500):GOTO840
840 ENDPROC
850 REM

```