Rubik’s Cube Demystified

By Curtis and Lillian J. Cooper

This program, in Applesoft BASIC using low-resolution graphics, simulates Rubik’s Cube. The problem is to take any arrangement of the cube and restore it to its pristine state.

In solving Rubik’s Cube, each face on the cube can turn clockwise or counterclockwise. In addition, different views of the cube are obtained by rotating the cube about axes through the center squares of the top and bottom faces, right and left faces, and front and back faces.

Program Notes

The program uses an array R, dimensioned to 5 by 5 by 5, as its representation for Rubik’s Cube. Colors are stored numerically as follows:

- Magenta 1
- Orange 9
- Blue 2
- Green 4
- White 15
- Yellow 13

The F face (see photo for face identification) is stored in the middle 3 by 3 squares where x = 1. The R face is stored in the middle 3 by 3 squares where y = 1. The D face is stored in the middle 3 by 3 squares where z = 1. Similarly, the B face is stored in the middle 3 by 3 squares where x = 5; the L face is stored in the middle 3 by 3 squares where y = 5; and the U face is stored in the middle 3 by 3 squares where z = 5.

Address correspondence to Curtis and Lillian J. Cooper, 803 E. Clark, Warrensburg, MO 64093.

---

Program listing, Rubik’s Cube simulation in Applesoft BASIC:

```
10 PRINT "THIS PROGRAM SIMulates RUBIK'S CUBE"
20 REM
30 REM INITIALIZE RUBIK'S CUBE
40 REM
50 DIM R(5,5,5), R(5,5,5), R(5,5), R(5,5), R(5,5)
60 FOR I = 1 TO 5: FOR J = 1 TO 5: FOR K = 1 TO 5
70 R(I,J,K) = 0
80 NEXT K, NEXT J, NEXT I
90 FOR I = 2 TO 4: FOR J = 2 TO 4
100 R(I,1,J) = 4*R(I,1,J) = 15*R(I,1,J) = 4
110 R(1,I,J) = 4*R(1,I,J) = 15*R(1,I,J) = 4
120 NEXT J, NEXT I
130 REM
140 REM MIX RUBIK'S CUBE
150 REM
160 INPUT "INPUT NUMBER OF MIXES": N
170 X = "" N
180 FOR I = 1 TO N
190 X = INT (6 * RND (1)) AND (1), Y = INT (3 * RND (1))
200 IF X > 0 THEN 220
210 X = "F": GOTO 310
220 IF X > 0 THEN 240
230 X = "F": GOTO 310
```

More...
The program contains two big subroutines. One subroutine is used to rearrange the cube. The following notation, similar to that in James F. Nourse's *The Simple Solution to Rubik's Cube*, is used by the program to change the cube.

**Summary of Moves**

R+—Turn R face one quarter turn clockwise
R—Turn R face one quarter turn counterclockwise
R2—Turn R face one half turn
F+—Turn F face one quarter turn clockwise
F—Turn F face one quarter turn counterclockwise
F2—Turn F face one half turn
L+—Turn L face one quarter turn clockwise
L—Turn L face one quarter turn counterclockwise
L2—Turn L face one half turn
D+—Turn D face one quarter turn clockwise
D—Turn D face one quarter turn counterclockwise
D2—Turn D face one half turn
U+—Turn U face one quarter turn clockwise
U—Turn U face one quarter turn counterclockwise
U2—Turn U face one half turn
B+—Turn B face one quarter turn clockwise
B—Turn B face one quarter turn counterclockwise
B2—Turn B face one half turn.

**MFR** where # is 1, 2, or 3—Rotate the cube about the axis passing through the center squares of the up [top] and down [bottom] faces. Move F to R face 1, 2, or 3 times.

**MFU** where # is 1, 2, or 3—Rotate the cube about the axis passing through the center squares of the right and left faces. Move F to U face 1, 2, or 3 times.

**MUR** where # is 1, 2, or 3—Rotate the cube about the axis passing through the center squares of the front and back faces. Move U to R face 1, 2, or 3 times.

Several moves can be performed on the cube by concatenating together any of the above moves.

The second subroutine draws the cube. Two views are displayed on the screen. The first view shows the corner formed by the up, right, and front faces as the corner closest to the viewer. The second view has the opposite corner formed by the down, left, and back faces closest to the viewer. The faces are each labeled (see photo).

The program begins by initializing the cube and asking how many moves you want it to make to mix up the cube. It is then randomly mixed the number of times specified and the resulting cube is displayed. The program asks you to input your move or moves. Invalid move entries are rejected and you are asked to reenter your move. If S is input, the program stops. Otherwise the resulting cube is displayed and another move is requested.

Enjoy exploring this color-graphics version of Rubik's Cube.

**References**


---

**Listing continued.**

240 IF X < 2 THEN 260
250 X$ = "$"; GOSUB 310
260 IF X < 3 THEN 280
270 X$ = "$"; GOSUB 310
280 IF X < 4 THEN 300
290 X$ = "B"; GOSUB 310
300 X$ = "0"
310 IF Y < 0 THEN 330
320 Y$ = "1"; GOSUB 360
330 IF Y < 1 THEN 330
340 Y$ = "-1"; GOSUB 360
350 Y$ = "2"
360 Z$ = Z$ + X$ + Y$
370 NEXT J
380 GOSUB 500
390 GOSUB 2050
400 NEXT K
410 REM CHANGE RUBIK'S CUBE
420 REM
430 INPUT "INPUT MOVE "; K$;
440 GOSUB 500
450 GOSUB 2050
460 GOSUB 450
470 REM
480 REM PERFORM MOVES
490 REM
500 FOR I = 1 TO 5; FOR J = 1 TO 5; FOR K = 1 TO 5
510 R(I,J,K) = R(I,J,K)
520 NEXT K
530 NEXT J
540 NEXT I
550 REM MOVE FACES
560 REM
570 IF X$ = "1" THEN 310
580 REM
590 REM
600 Y$ = MID$(Z$,2,1)
610 IF X$ < 1 THEN 710
620 IF I = 1 TO 5; FOR J = 1 TO 5
630 A(I,J) = R(I,J)B(I,J) = R(I,J)
640 NEXT J
650 GOSUB 1720
660 IF E = 1 THEN 1670
670 FOR I = 1 TO 5; FOR J = 1 TO 5
690 NEXT J
700 GOSUB 1200
710 IF X$ < 1 THEN 810
720 FOR I = 1 TO 5; FOR J = 1 TO 5
740 NEXT J
750 GOSUB 1800
760 IF E = 1 THEN 1670
770 FOR I = 1 TO 5; FOR J = 1 TO 5
780 R(I,J) = R(I,J)B(I,J) = B(I,J)
790 NEXT J
800 GOSUB 1200
810 IF X$ < 1 THEN 910
820 FOR I = 1 TO 5; FOR J = 1 TO 5
830 A(I,J) = R(I,J)B(I,J) = R(I,J)
840 NEXT J
850 GOSUB 1800
860 IF E = 1 THEN 1670
870 FOR I = 1 TO 5; FOR J = 1 TO 5
880 R(I,J) = R(I,J)B(I,J) = B(I,J)
890 NEXT J
900 GOSUB 1200
910 IF X$ < 1 THEN 1010
920 FOR I = 1 TO 5; FOR J = 1 TO 5
940 NEXT J
950 GOSUB 1720
960 IF E = 1 THEN 1670
970 FOR I = 1 TO 5; FOR J = 1 TO 5
980 R(I,J) = R(I,J)B(I,J) = B(I,J)
990 NEXT J
1000 GOSUB 1200
1010 IF X$ < 1 THEN 1110
1020 FOR I = 1 TO 5; FOR J = 1 TO 5
1030 A(I,J) = R(I,J)B(I,J) = R(I,J)
1040 NEXT J
1050 GOSUB 1720
1060 IF E = 1 THEN 1670
1070 FOR I = 1 TO 5; FOR J = 1 TO 5

---

Microcomputing, January 1982 29
Listing continued.
1090 NEXT J: NEXT I
1100 GOTO 1200
1110 IF X < "#" THEN 1670
1120 FOR I = 1 TO 5: FOR J = 1 TO 5
1130 A(I,J) = R(I,J)*I + B(I,J) = R(I,J,2)
1140 NEXT J: NEXT I
1150 GOSUB 1800
1160 IF E = 1 THEN 1670
1170 FOR I = 1 TO 5: FOR J = 1 TO 5
1180 R(I,J) = A(I,J)*R(I,J,2) - B(I,J)
1190 NEXT J: NEXT I
1200 X = MID$ (Z,3)
1210 GOTO 530
1220 REM
1230 REM CHANGE VIEWING CORNERS
1240 REM
1250 XX = MID$ (Z,2,1):YK = MID$ (Z,4,1)
1260 XK = MID$ (Z,4,1)
1270 IF XK < "1" AND XK > "2" AND YK < "1" AND YK > "3" THEN 1670
1280 N = VAL (2K)
1290 IF XK < "4" THEN 1540
1300 IF YK < "8" THEN 1420
1310 Z$ = MID$ (Z,5)
1320 FOR K = 1 TO N
1330 Z$ = "U-D-" + Z$ + "U"
1340 FOR I = 1 TO 5: FOR J = 1 TO 5
1350 A(I,J) = R(I,J,3)
1360 NEXT J: NEXT I
1370 FOR I = 1 TO 5: FOR J = 1 TO 5
1380 R(I,J) = A(I,J)
1390 NEXT J: NEXT I
1400 NEXT K
1410 GOTO 530
1420 IF YK < "U" THEN 1670
1430 Z$ = MID$ (Z,5)
1440 FOR K = 1 TO N
1450 Z$ = "F+H-" + Z$ + "F"
1460 FOR I = 1 TO 5: FOR J = 1 TO 5
1470 A(I,J) = R(I,J,3)
1480 NEXT J: NEXT I
1490 FOR I = 1 TO 5: FOR J = 1 TO 5
1500 R(I,J) = A(I,J)
1510 NEXT J: NEXT I
1520 NEXT K
1530 GOTO 530
1540 IF YK < "U" THEN 1670
1550 IF YK < "U" THEN 1670
1560 Z$ = MID$ (Z,5)
1570 FOR K = 1 TO N
1580 Z$ = "F+H-" + Z$ + "F"
1590 FOR I = 1 TO 5: FOR J = 1 TO 5
1600 A(I,J) = R(I,J,3)
1610 NEXT J: NEXT I
1620 FOR I = 1 TO 5: FOR J = 1 TO 5
1630 R(I,J) = A(I,J)
1640 NEXT J: NEXT I
1650 NEXT K
1660 GOTO 530
1670 PRINT "INVALID MOVE, TRY AGAIN."
1680 FOR I = 1 TO 5: FOR J = 1 TO 5: FOR K = 1 TO 5
1690 R(I,J,K) = R(I,J,3)
1700 NEXT K: NEXT J: NEXT I
1710 RETURN
1720 E = 0
1730 IF YK < "*" THEN 1750
1740 GOSUB 1950: RETURN
1750 IF YK < "=" THEN 1770
1760 GOSUB 1800: RETURN
1770 IF YK < "*" THEN 1790
1780 GOSUB 1800: GOSUB 1800: RETURN
1790 E = 11: RETURN
1800 E = 0
1810 IF YK < "=" THEN 1830
1820 IF YK < "=" THEN 1850
1830 RETURN
1840 GOSUB 1950: RETURN
1850 IF YK < "*" THEN 1790
1860 GOSUB 1800: GOSUB 1800: RETURN
1870 E = 11: RETURN
1880 FOR I = 1 TO 5: FOR J = 1 TO 5
1890 C(I,J) = A(I,J)*D(I,J) = B(I,J)
1900 NEXT J: NEXT I
1910 FOR I = 1 TO 5: FOR J = 1 TO 5
1920 C(I,J) = A(I,J)*D(I,J) = B(I,J)
1930 NEXT J: NEXT I
1940 RETURN
1950 FOR I = 1 TO 5: FOR J = 1 TO 5
1970 NEXT J: NEXT I
1980 FOR I = 1 TO 5: FOR J = 1 TO 5
2000 NEXT J: NEXT I
2010 RETURN
2020 REM
2030 REM PRINT RUBIK'S CUBE
2040 REM
2050 REM
2060 REM
2070 REM
2080 REM
2090 REM
2100 REM
2110 NEXT I