P2 - Pocket Cube

Solve a Pocket Cube (2x2x2 Rubik's Cube) with **minimum** number of moves and print the sequence of actions! (Hint: **Between BFS and DFS you should use the one that gives you the optimal path to the goal state!**)

There are 12 possible moves (only 90-degree moves are considered) in any state of the cube which are indicated in ***Figure 1*** with a specific number assigned to each move.

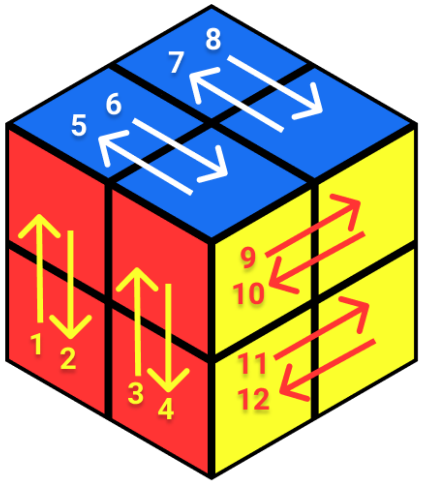

Figure 1 (front face: red, right face: yellow, top face: blue)

Input

The starting state (colors) for a cube is given in 6 separate lines according to the below order:

- L1 L2 L3 L4 (Left Face)
- U1 U2 U3 U4 (Top Face)
- F1 F2 F3 F4 (Front Face)
- D1 D2 D3 D4 (Bottom Face)
- R1 R2 R3 R4 (Right Face)

- B1 B2 B3 B4 (Back Face)

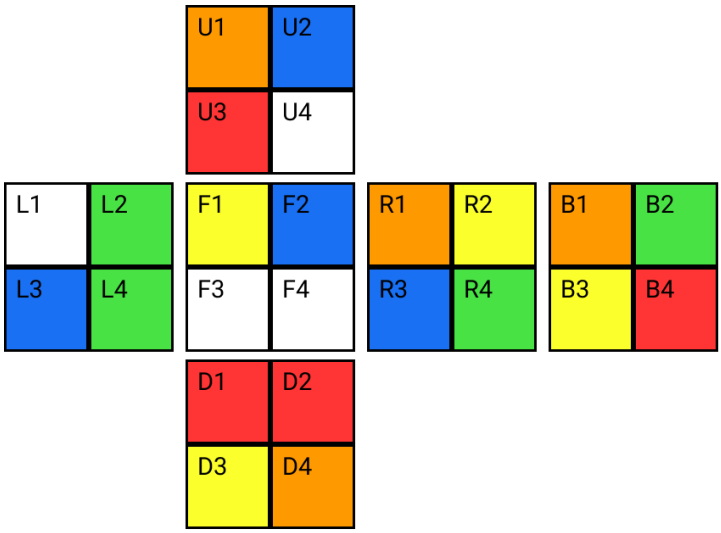

Figure 2

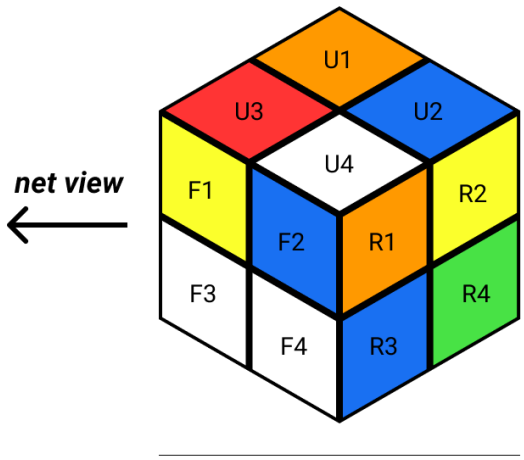

Figure 3

For instance, starting state for the cube displayed in ***Figure 2*** and ***Figure 3*** would be given as follows:

```
w g b g
o b r w
y b w w
r r y o
o y b g
o g y r
```

(w = white, g = green, b = blue, o = orange, r = red, y = yellow)

Output

The sequence of moves (according to the specific numbers assigned to each move in ***Figure 1***) that results in a solved cube should be printed in one line (space-separated).

The only accepted state for a solved cube is demonstrated in ***Figure 4***.

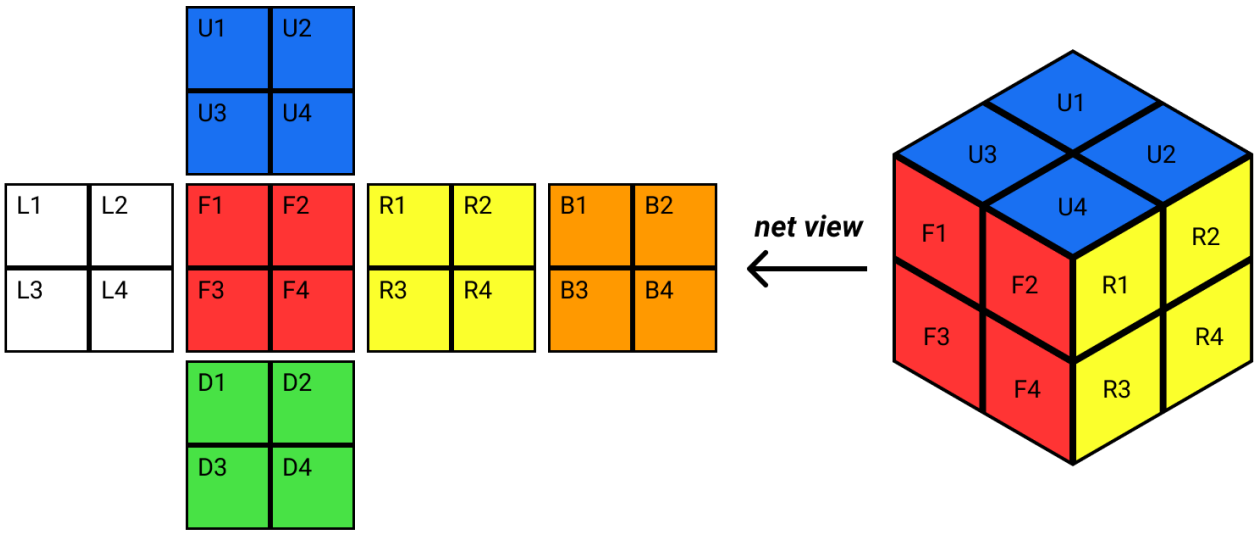

Figure 4 (the only accepted state for a solved cube)

Sample 1

Input:

```
w w w g
r b r r
g g y r
o y o g
y y b o
o b w b
```

Output:

```
5 2 11
```

The initial state of the sample 1 is demonstrated in ***Figure 5***.

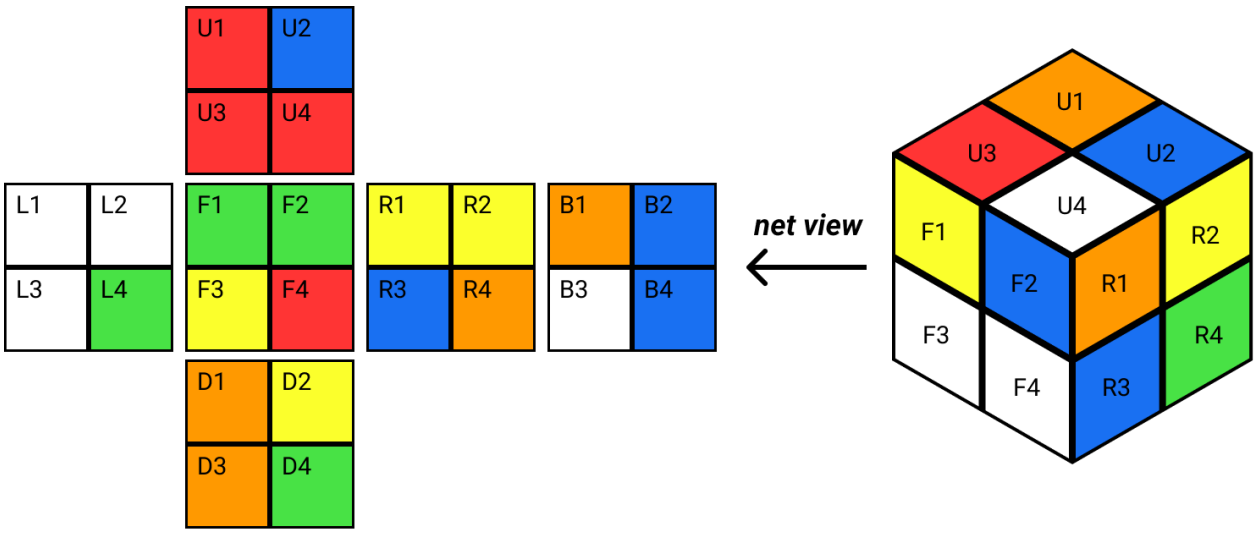

Figure 5

Then according to the output, state of the cube should be updated based on move number 5, 2 and 11 in order. Move number 5, updates the color of squares F1, F2, F3, F4, U3, U4, R1, R3, D1, D2, L2, L4. (**Figure 6**)

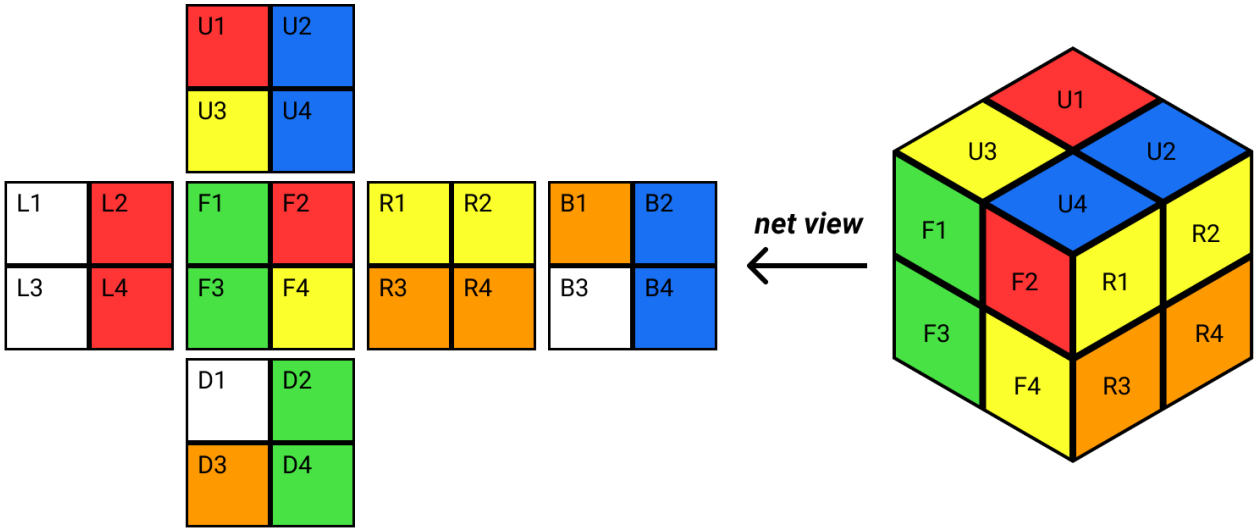

Figure 6

Next move, is move number 2, which updates color of squares L1, L2, L3, L4, B2, B4, U1, U3, F1, F3, D1, D3. (**Figure 7**)

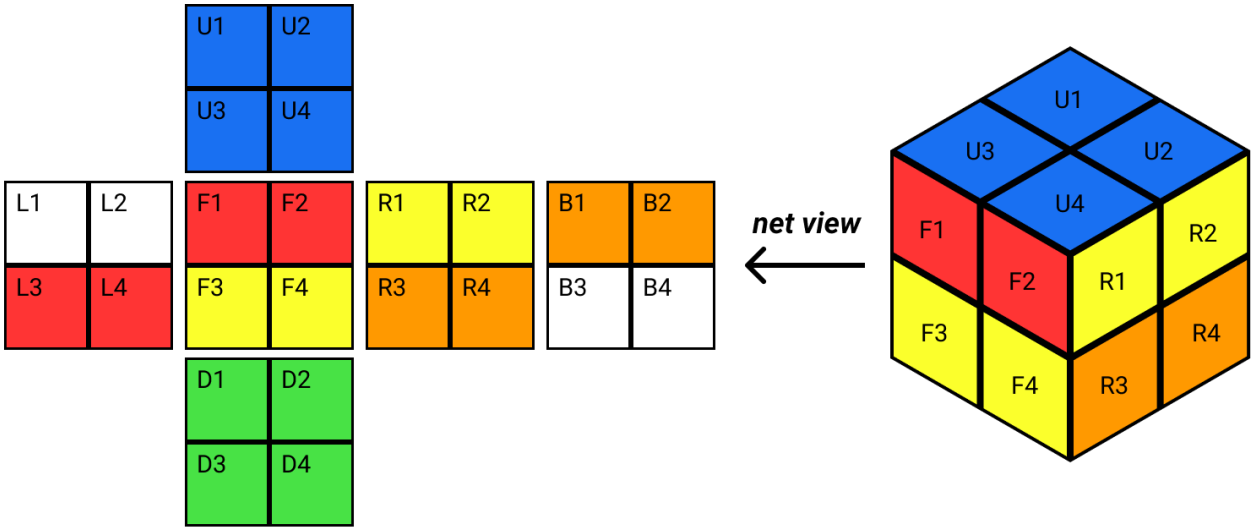

Figure 7

Final move, which solves the cube, is move number 11 and updates the color of squares D1, D2, D3, D4, L3, L4, F3, F4, R3, R4, B3, B4. (**Figure 8**)

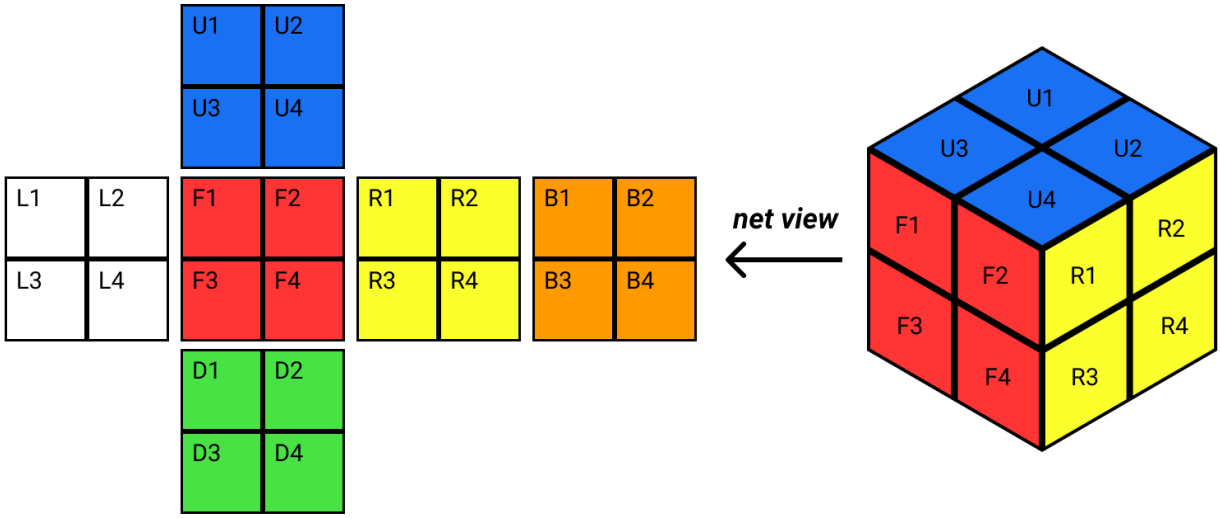

Figure 8

Sample 2

Input:

```
r b o g
g g y b
r w y y
o o w b
r r b w
w y o g
```

Output:

```
8 10 8 11 4
```

Note that depending on the order you use for expanding each node, your output might be different, but as long as the sequence of moves printed by your algorithm solves the cube and this is done in minimum number of moves, it is accepted.

For instance, another accepted output for the above sample is:

```
8 11 4 6 1
```

Also, cubes that are solvable with equal or less than 5 moves, should be solved in less than a few seconds (<5). (Your algorithm should be logically correct for any given test case, but due to the high branching factor of the problem, cubes that need 6 moves or more to be solved might take much longer time depending on your designed algorithm.)

Click **here** to view inputs of the test cases.

Click **here** to view outputs of the test cases.