

Rubiks Cube

Author: Courtney

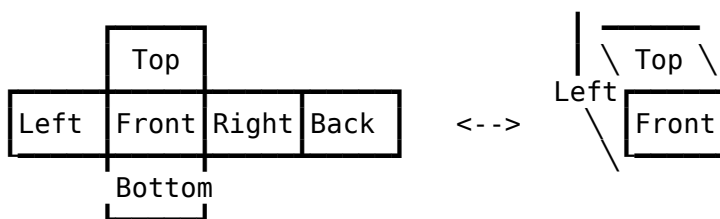
Instruction

Given the current state of a [Rubik's Cube](#) and a sequence of operations, find the state of the cube after those operations.

The input will be a JSON object with field **state** and **ops**. Compute the changes and return the final state of the cube.

The operations field **ops** is a string comprises of all the operations without any separations. For each face, there are 2 possible operations: rotate clockwise or anticlockwise. The anticlockwise operations will follow by an **i**. As there are 6 faces in the cube, there are 12 operations: **U, Ui, L, Li, F, Fi, R, Ri, B, Bi, D, Di** for the faces in the Up, Left, Front, Right, Back, Down direction respectively. The observation perspective never changes once the state of the cube is defined.

The **state** is another object with 6 fields representing the 6 faces of the cube using the first character of the face in lower case (aka **u, l, f, r, b, d**). Each face is represented in a 3 by 3 integer array from top-left to bottom-right as if the cube is being unfolded to the following 2D structure.



R = Right Face - Right side of the cube



L = Left Face - Left side of the cube



U = Up Face - Top side of the cube



D = Down Face - Bottom side of the cube



F = Front Face - Front side of the cube



B = Back Face - Back side of the cube



Endpoint

Expose a POST endpoint `/rubiks` in your server for grading on this question. Our server will POST the JSON input and expecting a JSON output within 1000ms.

Sample Input

```
{
  "ops": "UiD",
  "state": {
    "u": [[0, 0, 0], [0, 0, 0], [0, 0, 0]],
    "l": [[1, 1, 1], [1, 1, 1], [1, 1, 1]],
    "f": [[2, 2, 2], [2, 2, 2], [2, 2, 2]],
    "r": [[3, 3, 3], [3, 3, 3], [3, 3, 3]],
    "b": [[4, 4, 4], [4, 4, 4], [4, 4, 4]],
    "d": [[5, 5, 5], [5, 5, 5], [5, 5, 5]]
  }
}
```

Sample Output

```
{
  "u": [[0, 0, 0], [0, 0, 0], [0, 0, 0]],
  "l": [[4, 4, 4], [1, 1, 1], [4, 4, 4]],
  "f": [[1, 1, 1], [2, 2, 2], [1, 1, 1]],
  "r": [[2, 2, 2], [3, 3, 3], [2, 2, 2]],
  "b": [[3, 3, 3], [4, 4, 4], [3, 3, 3]],
  "d": [[5, 5, 5], [5, 5, 5], [5, 5, 5]]
}
```

Explanation

Rotate the up face anticlockwise to get

```
{
  "u": [[0, 0, 0], [0, 0, 0], [0, 0, 0]],
  "l": [[4, 4, 4], [1, 1, 1], [1, 1, 1]],
  "f": [[1, 1, 1], [2, 2, 2], [2, 2, 2]],
  "r": [[2, 2, 2], [3, 3, 3], [3, 3, 3]],
  "b": [[3, 3, 3], [4, 4, 4], [4, 4, 4]],
  "d": [[5, 5, 5], [5, 5, 5], [5, 5, 5]]
}
```

Then rotate the down face clockwise to get the final result.

Rules

0 <= Number of operations, Face values of the grid <= 100
Return the result within 1000ms

Scoring

Total 20 test cases, 5% of total score for each correct result