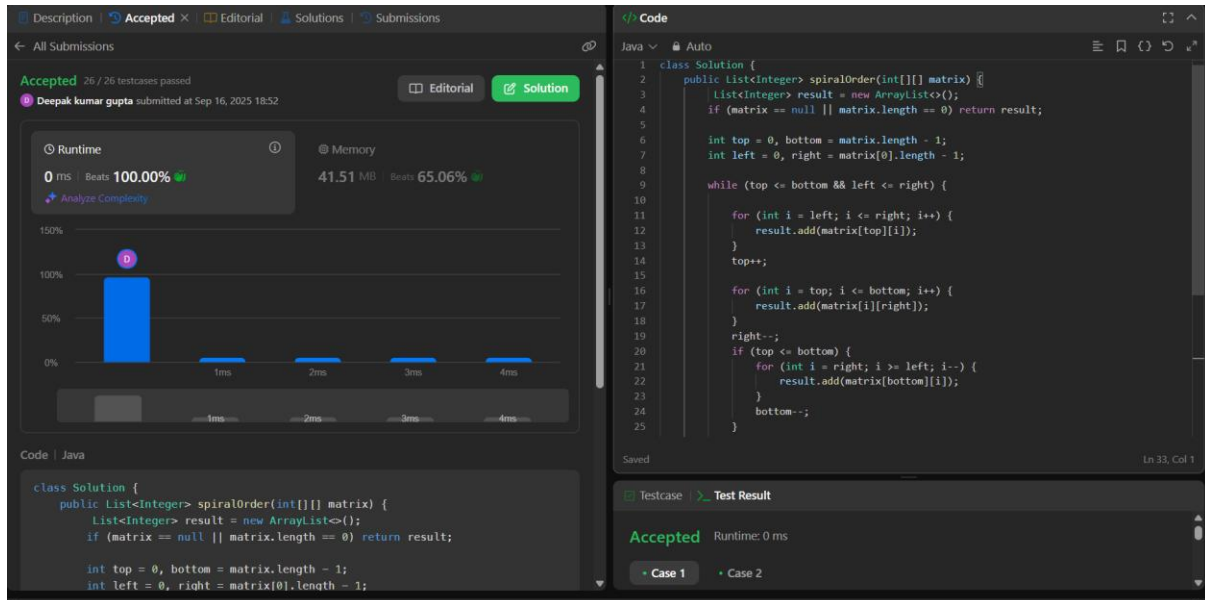


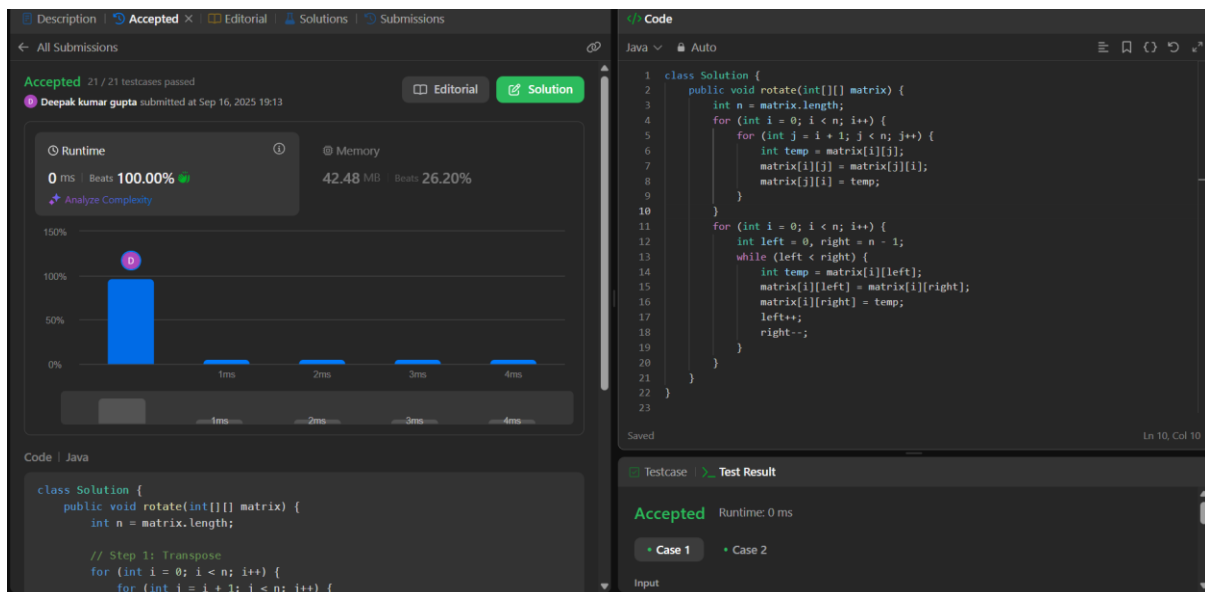
Leetcode:-

Problem No. :- 54



Leetcode:-

Problem No. :- 48



geeksforgeeks:-

problem(Rotate by 90 degree)

The screenshot shows the GeeksforGeeks interface for the problem "Rotate by 90 degree". The left sidebar displays the "Output Window" with "Compilation Results" showing "Problem Solved Successfully". It also shows "Test Cases Passed: 1120 / 1120", "Attempts: Correct / Total: 2 / 2", "Accuracy: 100%", and "Time Taken: 1.56". The right pane shows the Java code for the solution, which uses a temporary matrix to rotate the input matrix by 90 degrees.

```
1 class Solution {
2     public void rotateMatrix(int[][] mat) {
3         int n = mat.length;
4         for (int i = 0; i < n; i++) {
5             for (int j = i + 1; j < n; j++) {
6                 int temp = mat[i][j];
7                 mat[i][j] = mat[j][i];
8                 mat[j][i] = temp;
9             }
10        }
11    }
12    public void rotateMatrix(int[][] mat) {
13        int top = 0, bottom = n - 1;
14        while (top < bottom) {
15            int temp = mat[top][col];
16            mat[top][col] = mat[bottom][col];
17            mat[bottom][col] = temp;
18            top++;
19            bottom--;
20        }
21    }
22 }
23
24
```

geeksforgeeks:-

problem(Rotate a Matrix by 180
Counterclockwise)

The screenshot shows the GeeksforGeeks interface for the problem "Rotate a Matrix by 180 Counterclockwise". The left sidebar displays the "Output Window" with "Compilation Results" showing "Problem Solved Successfully". It also shows "Test Cases Passed: 1120 / 1120", "Attempts: Correct / Total: 1 / 2", "Accuracy: 50%", "Points Scored: 4 / 4", and "Time Taken: 1.93". The right pane shows the Java code for the solution, which rotates the matrix by 180 degrees counter-clockwise by swapping elements across the center.

```
1 class Solution {
2     public void rotateMatrix(int[][] mat) {
3         // code here
4         int n = mat.length;
5         for (int i = 0; i < n; i++) {
6             for (int j = 0; j < n; j++) {
7                 int x = n - 1 - i;
8                 int y = n - 1 - j;
9                 if (i < x || (i == x && j < y)) {
10                    int temp = mat[i][j];
11                    mat[i][j] = mat[x][y];
12                    mat[x][y] = temp;
13                }
14            }
15        }
16    }
17 }
18
```

