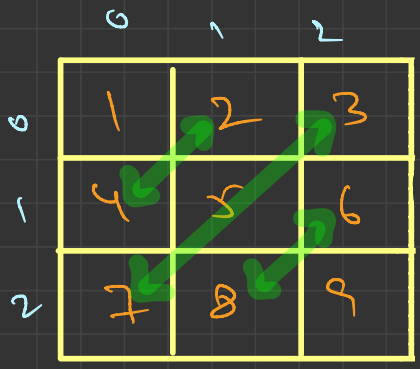
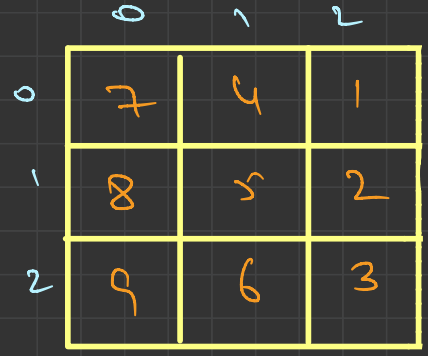
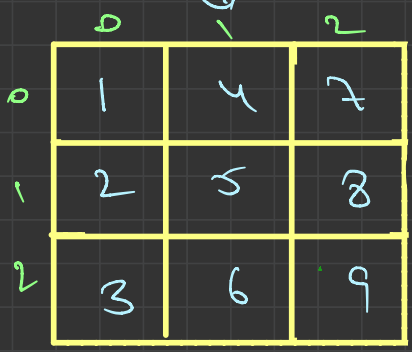


Input



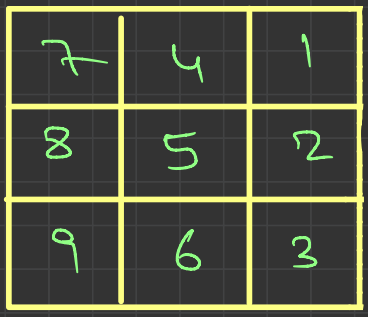
Output

① ↓



Transpose

② →



Reverse col

```
class Solution {
public:
    void rotate(vector<vector<int>>& matrix) {
        int n = matrix.size();

        // transpose
        for(int row = 0; row < n; row++){
            for(int col = 0; col < row; col++){
                swap(matrix[row][col],matrix[col][row]);
            }
        }

        // reverse col wise
        for(int row = 0; row < n; row++){
            for(int col = 0; col < n/2; col++){
                swap(matrix[row][col], matrix[row][n-col-1]);
            }
        }


    }
};
```

using manual reverse

```
class Solution {
public:
    void rotate(vector<vector<int>>& matrix) {
        int n = matrix.size();

        // transpose
        for(int row = 0; row < n; row++){
            for(int col = 0; col < row; col++){
                swap(matrix[row][col],matrix[col][row]);
            }
        }

        // reverse col wise
        for(int row = 0; row < n; row++){
            reverse(matrix[i].begin(), matrix[i].end());
        }
    }
};
```

using reverse stl in 2d vectors