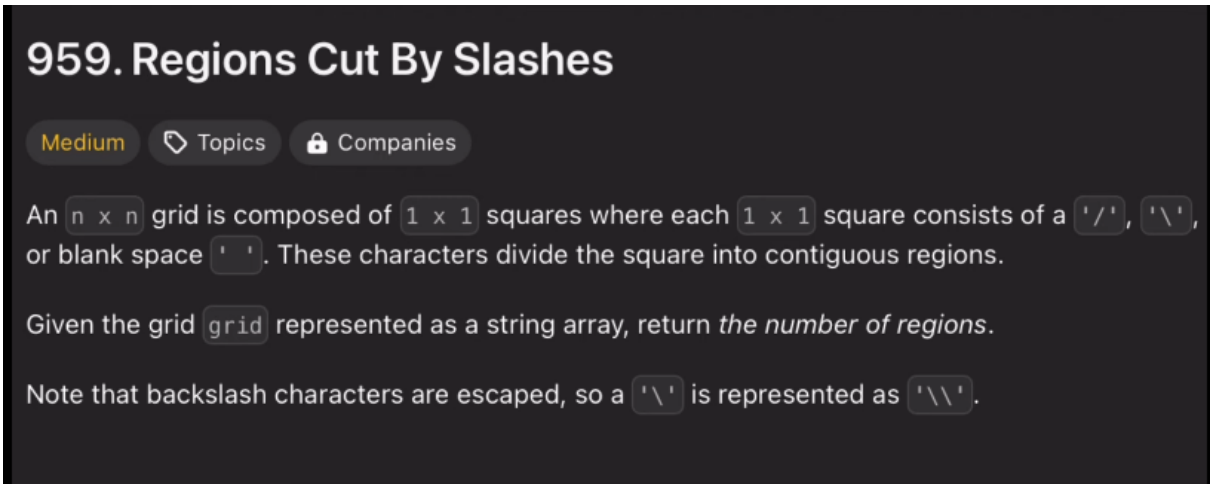
959 Regions Cut By Slashes - 10/08/24 (medium) (Google)

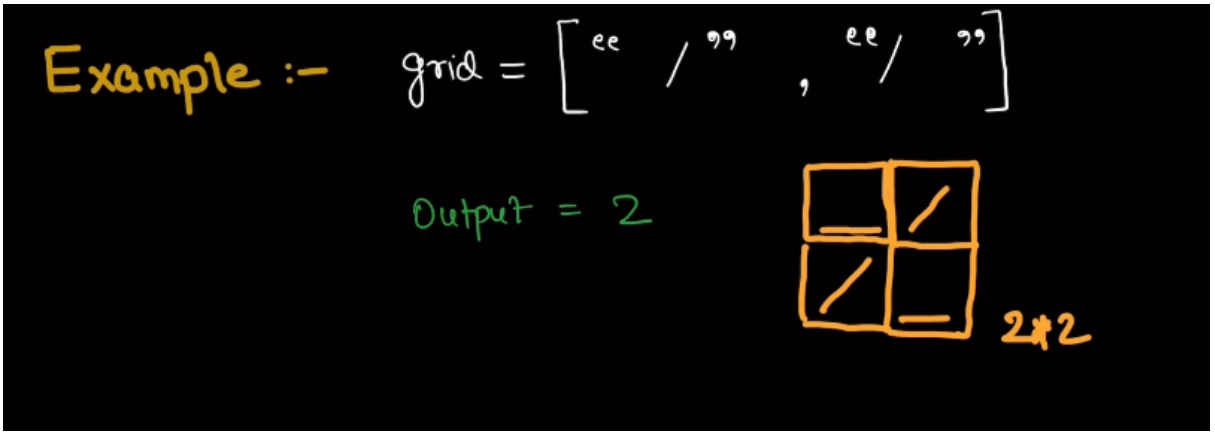
Google Asked



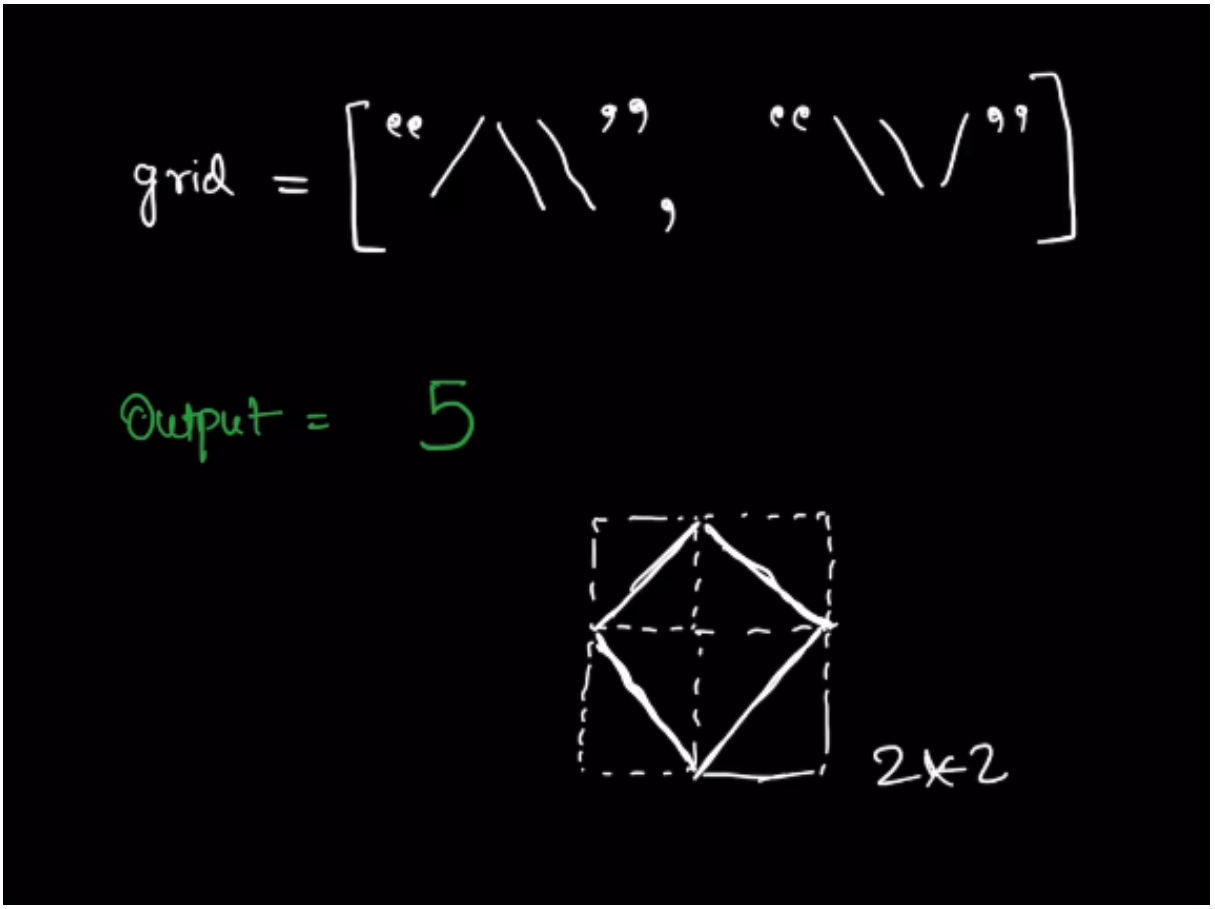
"\" is treated as escaped character by compiler

then if you want '\' then use '\\'

Example 1:




```
gird = [" /","/ "]
```

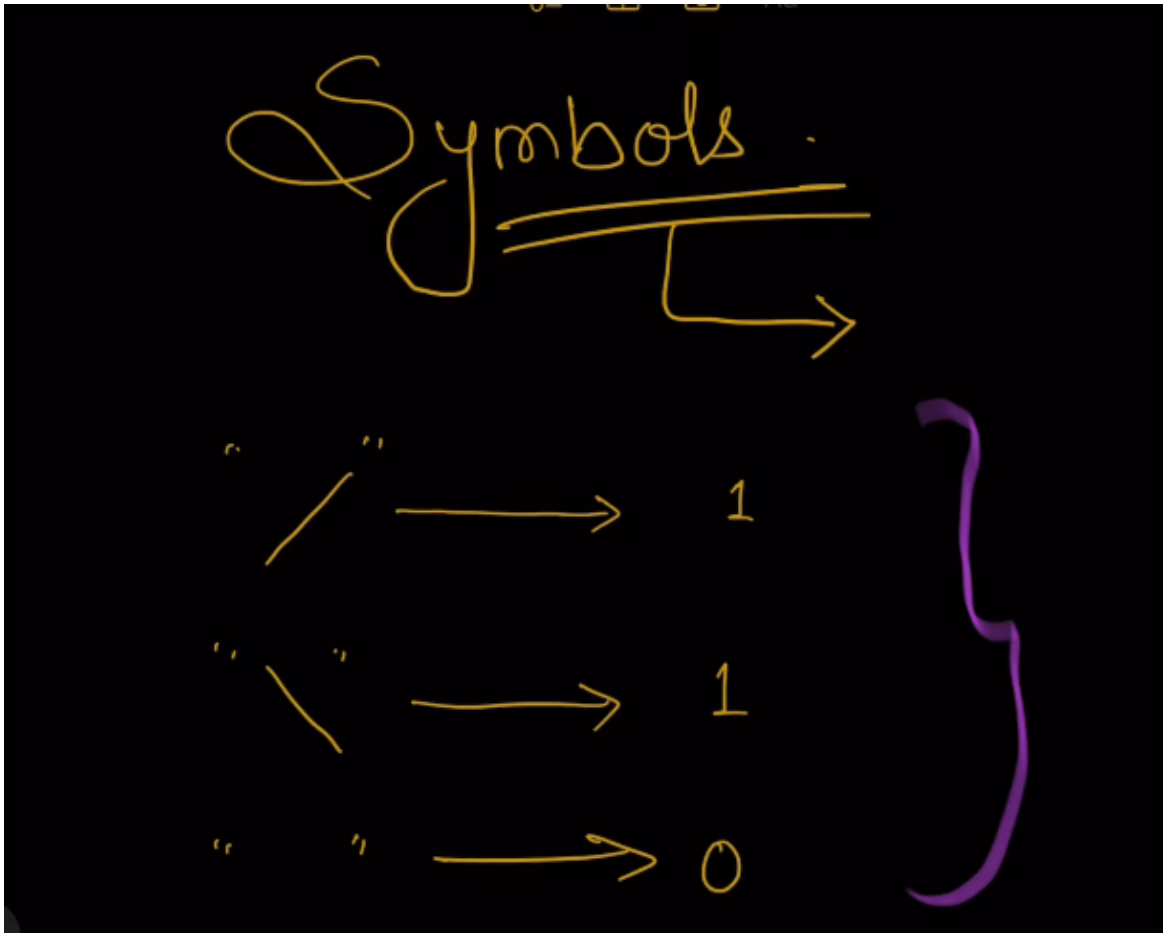



```
gird = ["/\\","\\/"]
```

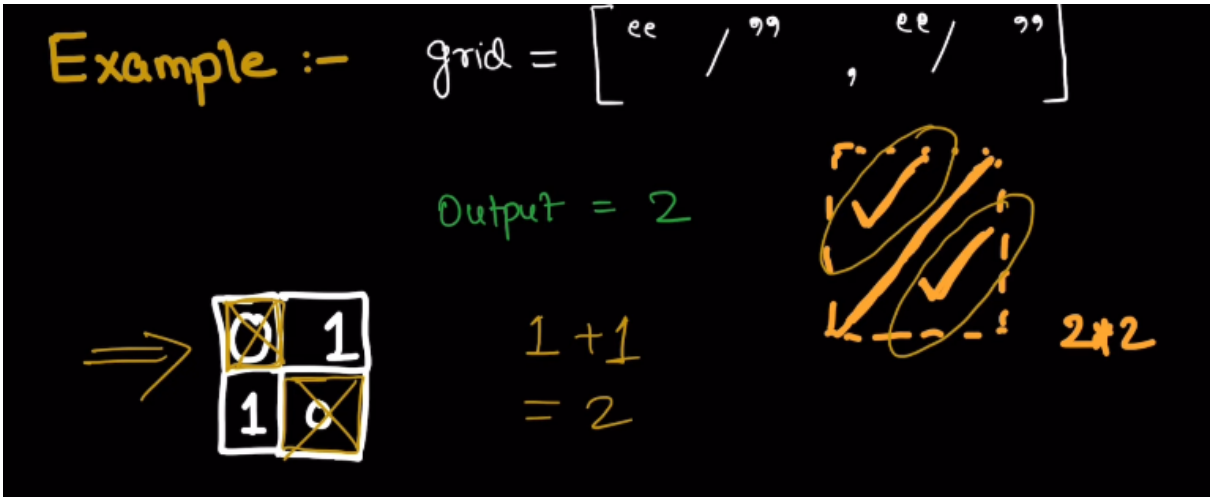
Tips:

when symbol related probkem come

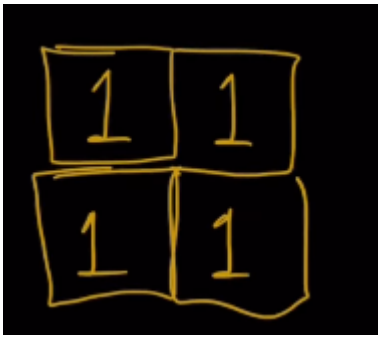
try karo ki - kya isko me aise form me represent kar sakta hu numberical me



now try using 0 and 1

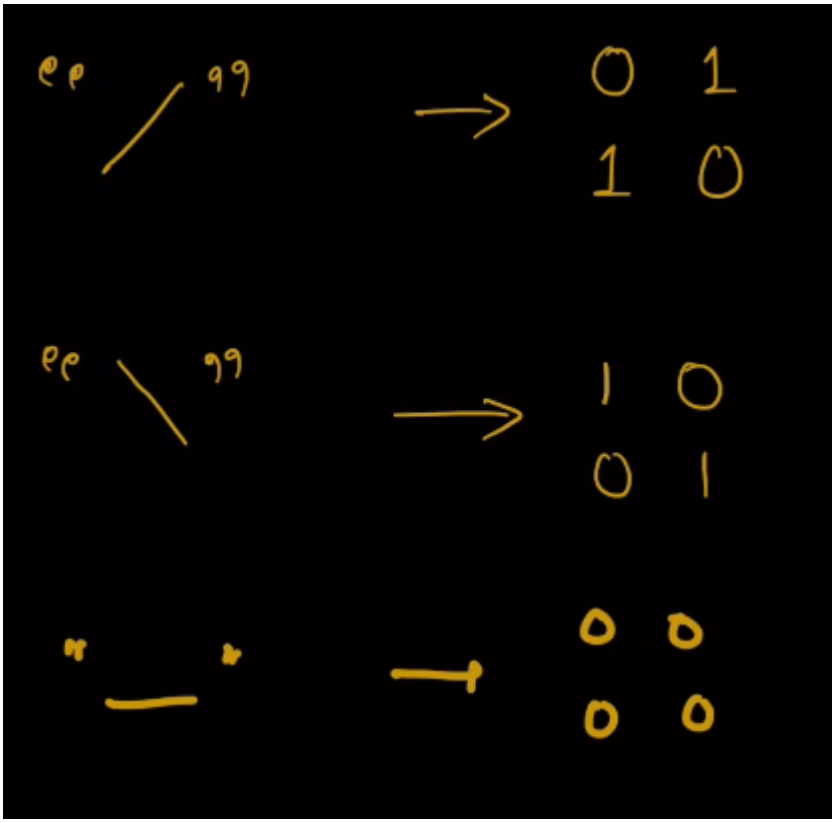


here we test DFS on 0 if there is no another 0 on right , left, uop and down
then make it visited and +1 it



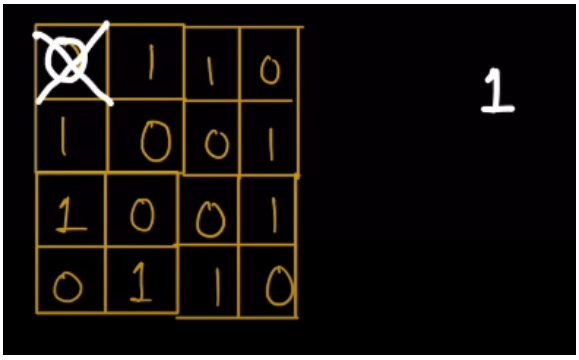
but there is the probelm if i slect all symbol as 1

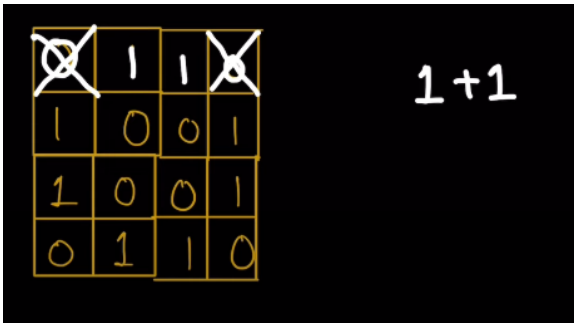
NOW choose another symbol notation

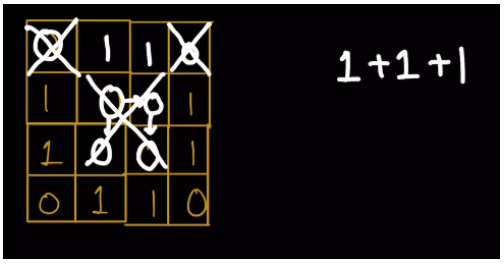


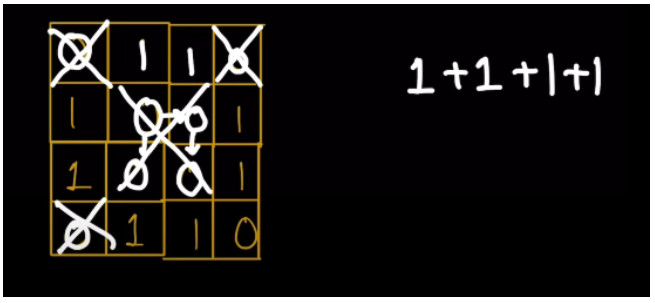
Now finding pattern for

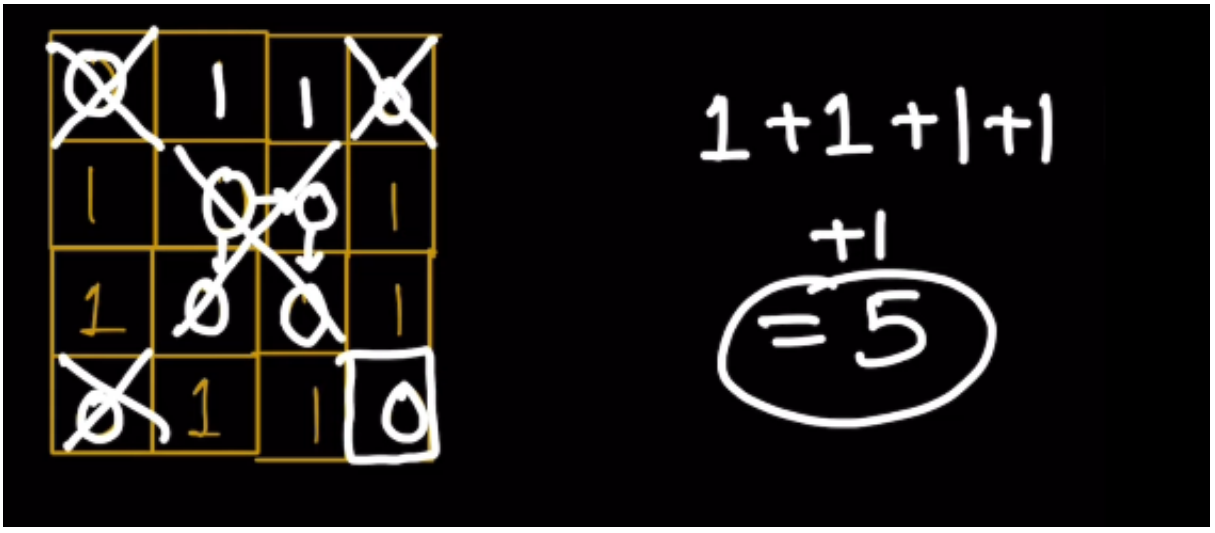
```
gird = ["/\\","\\/"]
```





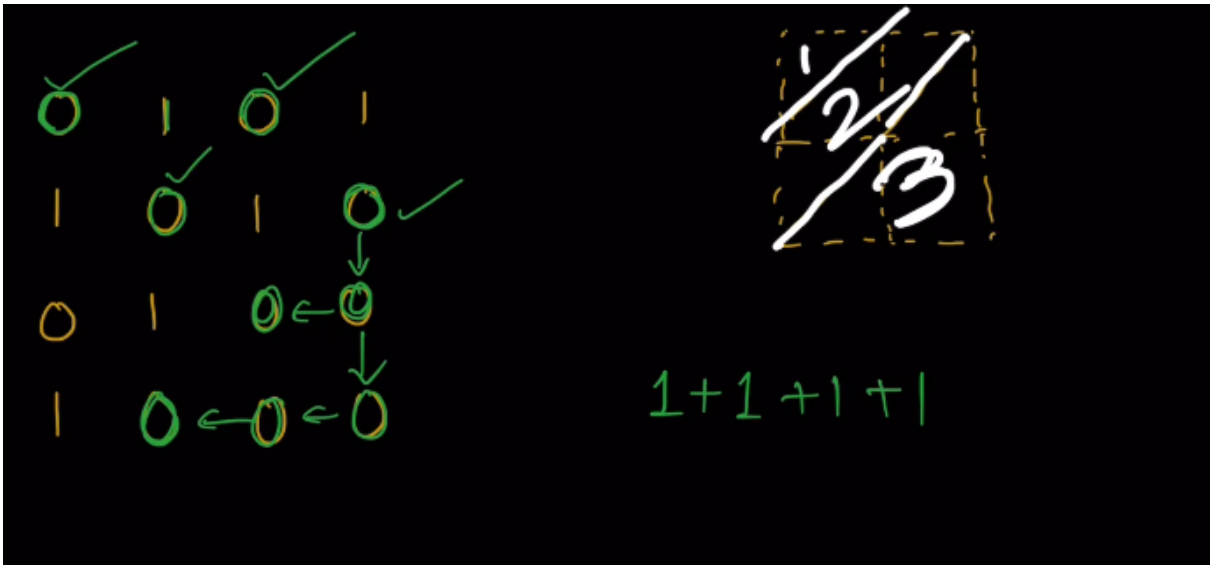






now to check for this

```
gird = ["//","/ "]
```

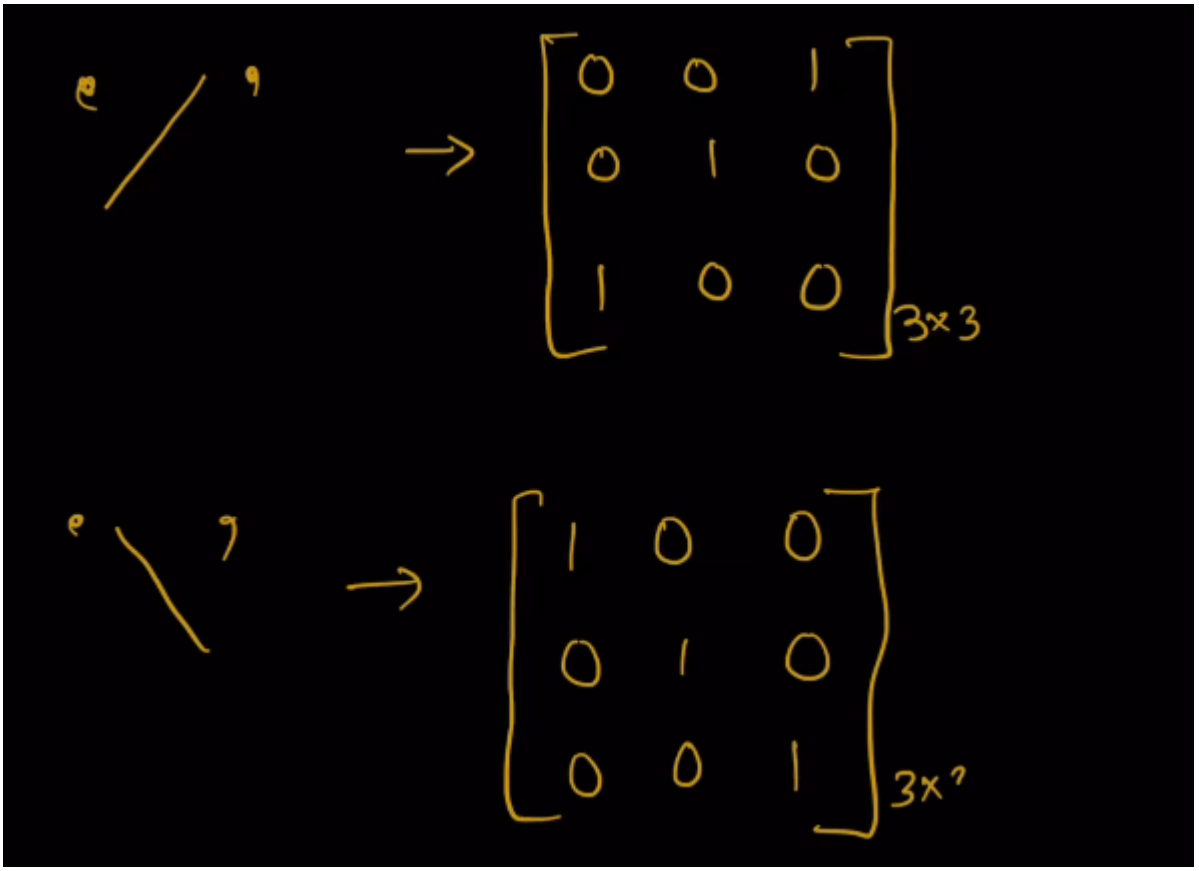


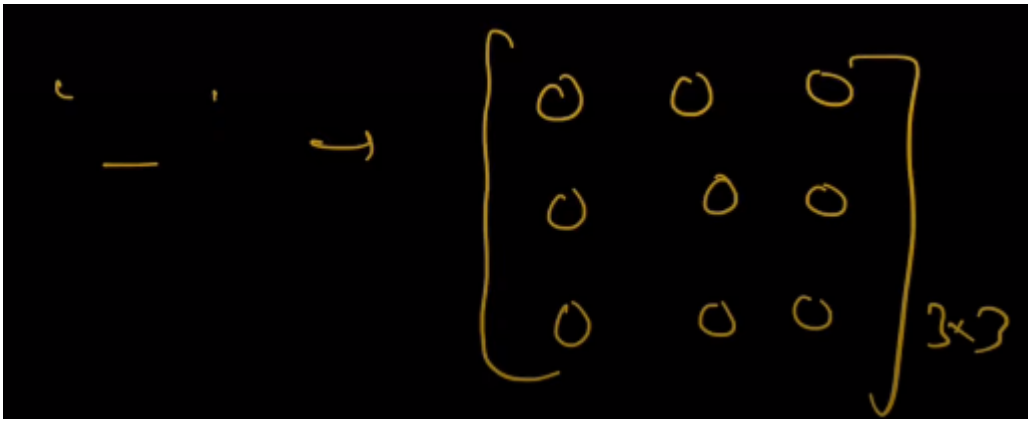
but it failed here

because we only have 3 region , but it has 5 region which is not acceptable

This also fail to represent when 0 or “ “ comes

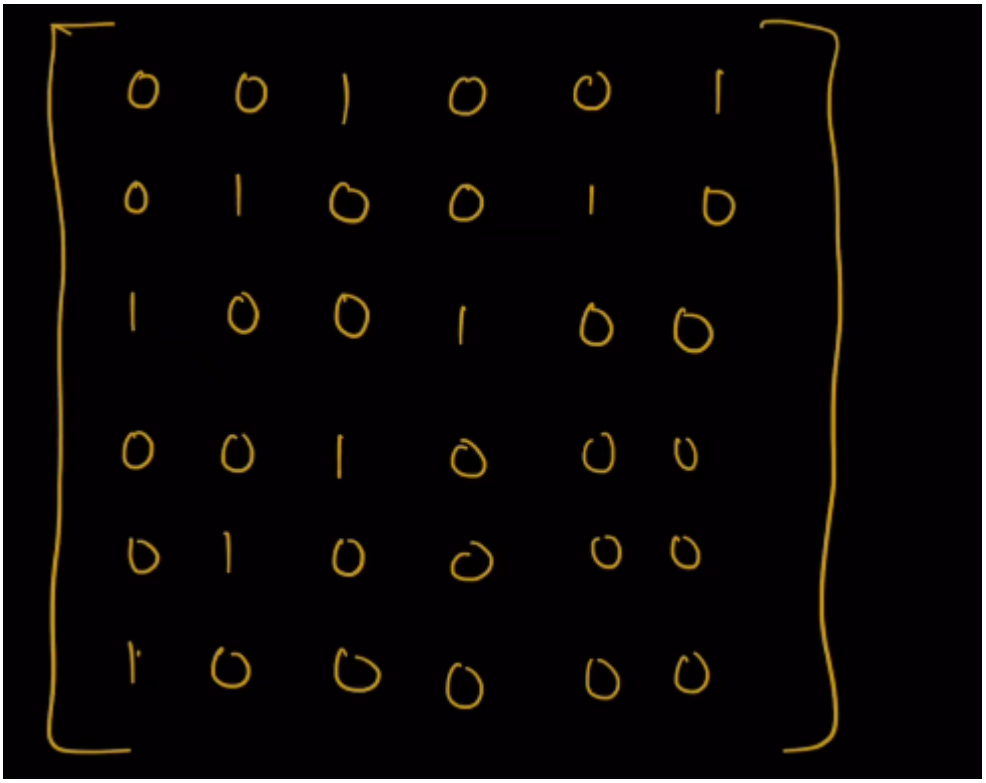
WE will Try 3x3 pattern now



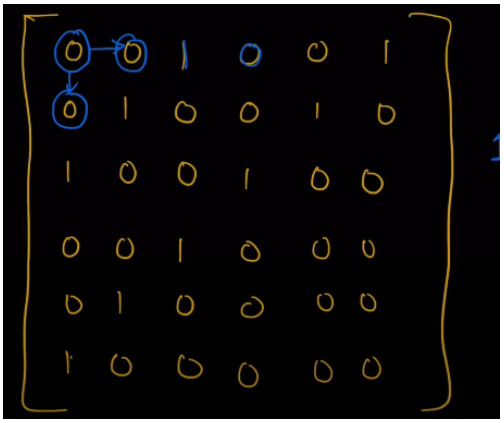


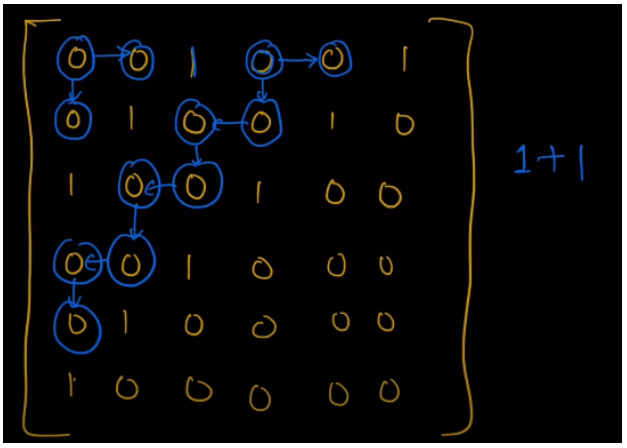
now to check for this

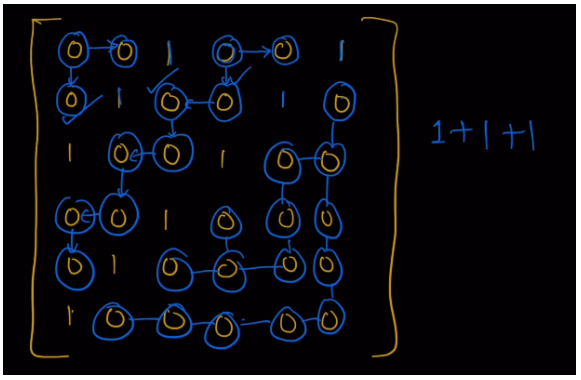
```
gird = ["//","/ "]
```

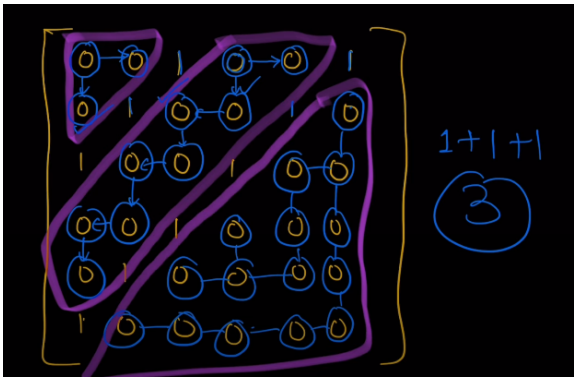


LETS START





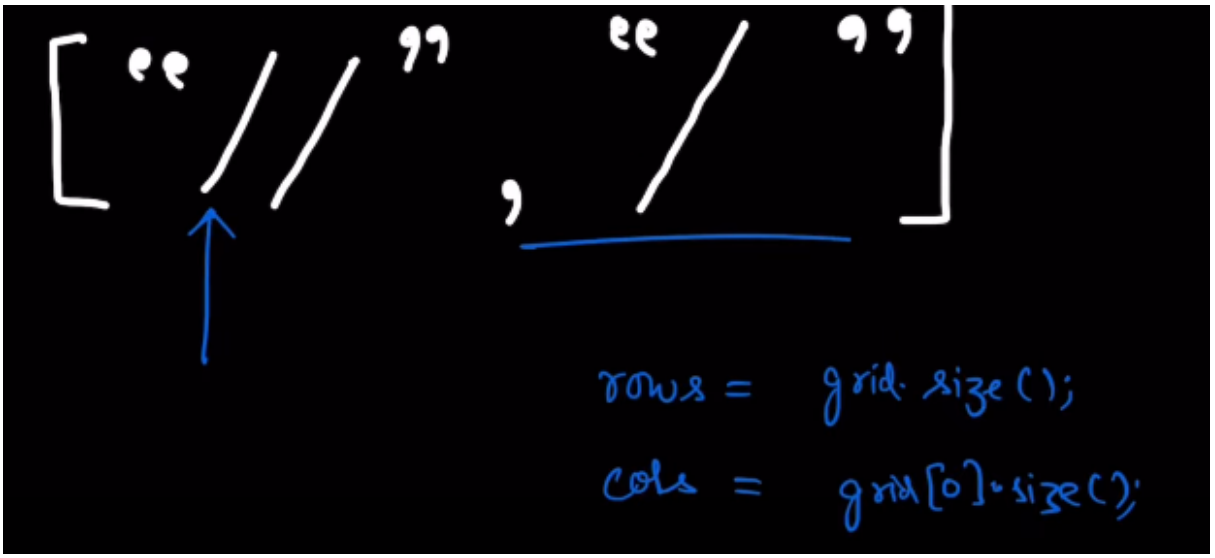


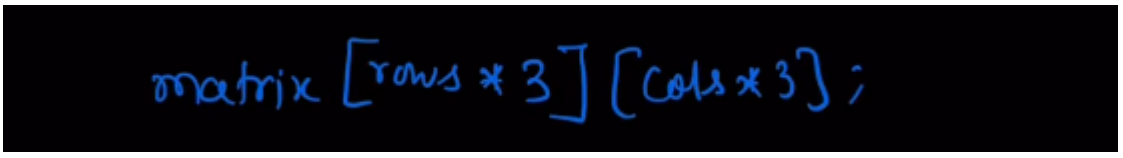


this work completely fine

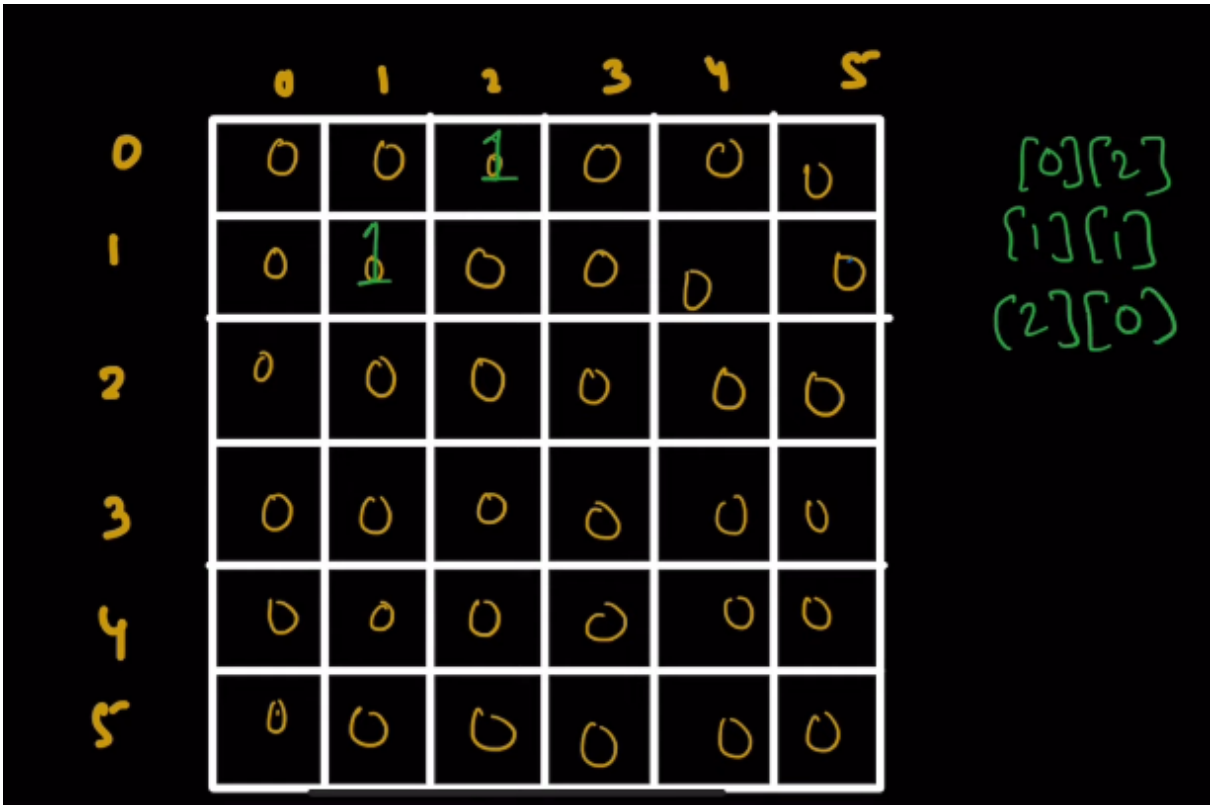
so we will use 3×3 pattern

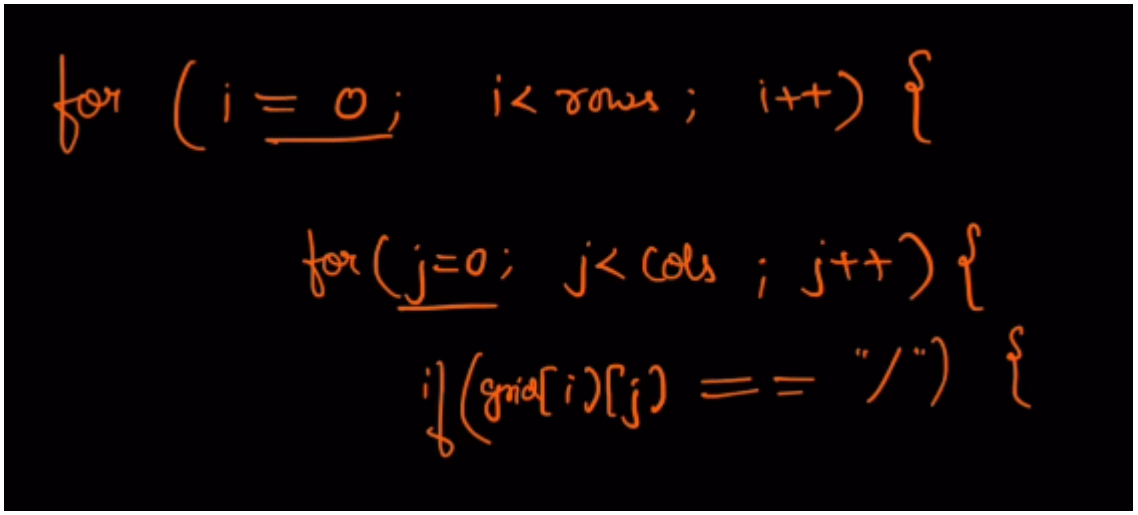
Coding it

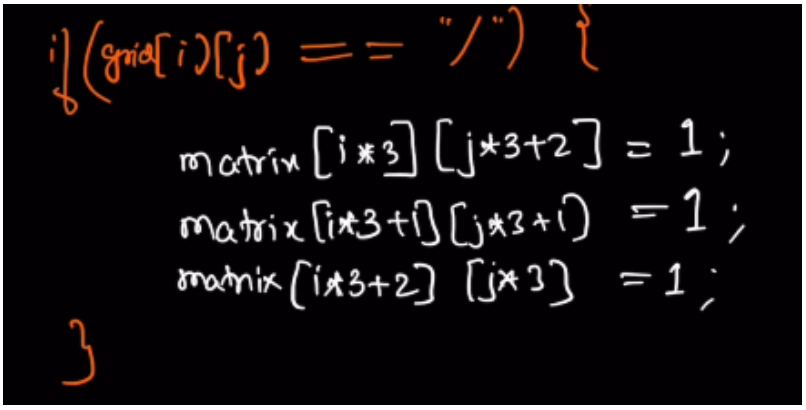




for “/”

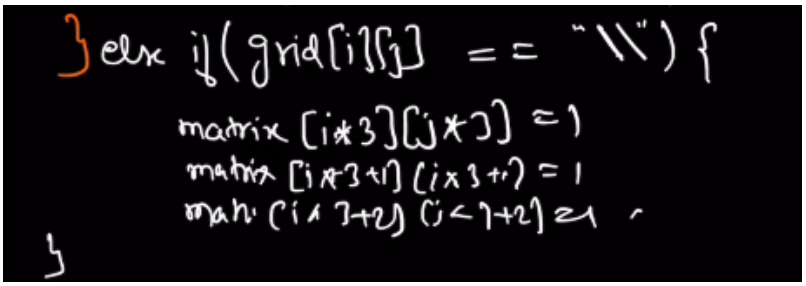

0 1 2 3 4 5
0 0 0 1 0 0 0
1 0 1 0 0 0 0
2 0 0 0 0 0 0
3 0 0 0 0 0 0
4 0 0 0 0 0 0
5 0 0 0 0 0 0
[0][2]
[1][1]
[2][0]


for (i = 0; i< rows ; i++) {
for(j=0; j< cols ; j++) {
if(grid[i][j] == "/") {

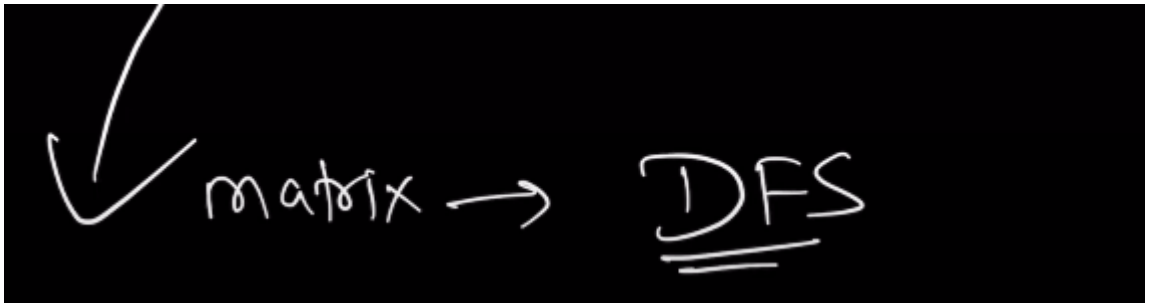

if(grid[i][j] == "/") {
matrix[i*3][j*3+2] = 1;
matrix[i*3+1][j*3+1] = 1;
matrix[i*3+2][j*3] = 1;
}


[0][2]
[1][1]
[2][0]

for “\\”



To find “0” or “ “



Solution

```
class Solution {
public:

    void numberOfIslandDFS(vector<vector<int>>& matrix, int i
        if(i<0 || i>=matrix.size() || j<0 || j>=matrix[0].siz
            return ;
        }

        matrix[i][j] =1;
        numberOfIslandDFS(matrix,i+1,j);
```

```
        numberOfIslandDFS(matrix,i-1,j);
        numberOfIslandDFS(matrix,i,j+1);
        numberOfIslandDFS(matrix,i,j-1);
    }

    int regionsBySlashes(vector<string>& grid) {
        int rows = grid.size();
        int cols = grid[0].size();
        int region =0;
     // initialize the matri with zero so writen 0i here
        vector<vector<int>> matrix(rows*3, vector<int>(cols*3

        for (int i=0;i<rows;i++){
            for(int j=0;j<cols;j++){
                if(grid[i][j]=='/'){
                    matrix[i*3][j*3+2]=1;
                    matrix[i*3+1][j*3+1]=1;
                    matrix[i*3+2][j*3]=1;
                }
                else if(grid[i][j]=='\\'){
                    matrix[i*3][j*3]=1;
                    matrix[i*3+1][j*3+1]=1;
                    matrix[i*3+2][j*3+2]=1;
                }
            }
        }

        //apply concept of island porblem
        for(int i=0;i<matrix.size();i++){
            for(int j=0;j<matrix[0].size();j++){
                if(matrix[i][j]==0){
                    numberOfIslandDFS(matrix,i,j);
                    region+=1;
                }
            }
        }
        return region;
```

```
    }
};
```