📑 Description △ Solution 💬 Discuss (435) 🕐 Submissions

ⓘ C++ ▾ Autocomplete ⓘ {} ↺

Success   Details ›

Runtime: **44 ms**, faster than **98.07%** of C++ online submissions for Number of Enclaves.

Memory Usage: **21.6 MB**, less than **87.02%** of C++ online submissions for Number of Enclaves.

Next challenges:

(Find First and Last Position of Element in Sorted Array)   (Clone Graph)

(Tree of Coprimes)

Show off your acceptance: 🅕 🐦 in

Time Submitted	Status	Runtime	Memory	Language
11/29/2021 08:13	Accepted	44 ms	21.6 MB	cpp

```cpp
class Solution {
public:
    void dfs(vector<vector<int>>&grid,int i,int j){
        int r=grid.size();
        int c=grid[0].size();

        if(i<0||j<0||i>=r||j>=c||grid[i][j]!=1) //base condition
            return;

        grid[i][j]=2;

        dfs(grid,i+1,j);
        dfs(grid,i-1,j);
        dfs(grid,i,j+1);
        dfs(grid,i,j-1);
    }
    int numEnclaves(vector<vector<int>>& grid){
        int r= grid.size();
        int c= grid[0].size();

        for(int i=0;i<r;i++){
            for(int j=0;j<c;j++){
                if(i==0 || j==0 || i==r-1 || j==c-1)
                    dfs(grid,i,j);
                }
            }
        int res=0; //increment count everytime you see 1
        for(int i=0;i<r;i++){
            for(int j=0;j<c;j++){
                if(grid[i][j]==1)
                    res++;
                }
            }
        return res;
    }
```