Aeroplane bombing

Tuesday, July 22, 2025 9:31 PM

https://www.hackerrank.com/contests/target-samsung-13-nov19/challenges/aeroplane-bombing/problem

1	2	2	1	0
1	0	2	1	0
2	2	1	0	1
2	1	0	1	2

→ (N-1)th row theke start করলে)

শুরু করার পর এই 3টার যেকোন 1 টা থেকে start করতে পারবে।

Spaceship এর কাছে available move, 3, (↖, ↑, ↗)

```
f (N-1, col-1)
f (N-1, col+1)
f (N-1, col)

for (int d=-1; d<=1; d++)
  { f(N-1, col+d)
```

If I encounter bomb.

I check if I have used up my bomb. (bomb use করলে return)

If I have not, then make the bomb, 0 of the upper 5x5 box. of that, spaceship

then after the recursion is completed. revert it back to original

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;
int maxCnt = 0;
void solve(vector<vector<int>>& field,vector<vector<int>>& copyField,int 
currCount,int row,int col,int bombed)
{
    //jodi top row er baire chole jai
    if(row < 0 || col< 0 || col >= 5) return;
    if(field[row][col] == 1 || field[row][col] == 0)
    {
        currCount += field[row][col];
        maxCnt = max(currCount,maxCnt);
        for(int d = -1;d <=1;d+=1)
        {
            solve(field,copyField,currCount,row-1,col+d,bombed);
        }
    }

    else if(field[row][col] == 2)
    {
        if(bombed == 0)
        {
            return;
        }
        else
        {
            bombed = 0;
            for(int i = row;i>=0 && i >= row-4;i--)
            {
                for(int j = col-2;j<=col+2;j++)
                {
                    if(j>=0 && j<=4){
                        if(field[i][j] == 2)field[i][j] = 0;
                    }
                }
            }
           for(int d = -1;d <=1;d+=1)
           {
            solve(field,copyField,currCount,row-1,col+d,bombed);
           }

            for(int i = row;i>=0 && i >= row-4;i--)
            {
                for(int j = col-2;j<=col+2;j++)
                {
                    if(j>=0 && j<=4){
                        field[i][j] = copyField[i][j];
                    }
                }
            }

        }
    }

}
int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int t;
    cin>>t;
    for(int i = 1;i<=t;i++){
        int N;
        cin>>N;
        vector<vector<int>>field(N,vector<int>(5,0));
        vector<vector<int>>copyField(N,vector<int>(5,0));

        for(int i = 0;i<N;i++)
        {
            for(int j = 0;j<5;j++)
            {
                cin>>field[i][j];
                copyField[i][j] = field[i][j];
            }
        }
        maxCnt = 0;
        int currCount = 0;
        int col = 2;
        int bombed = 1;
        solve(field,copyField,currCount,N-1,col,bombed);
        solve(field,copyField,currCount,N-1,col+1,bombed);
        solve(field,copyField,currCount,N-1,col-1,bombed);

        cout << "#" << i << " " << maxCnt << endl;
    }
}
```