Description | Hints | Submissions | Discussions | Notes

[illegible]	1 sec	256000KB	100
Difficulty	Time Limit	Memory	Score

80/80 XP | 30/30

Description

Monkey D. Luffy, on his journey of becoming the *"King of Pirates"* and to conquer the *"One Piece"*, wants to travel across the Grand Line. Grand Line is a mysterious sea, and is in the shape of a $N * M$ grid S with each cell denoting the wind direction. The sign of $S[i][j]$ can be:

- 1 which means wind in the cell flows to the right. (i.e from $S[i][j]$ to $S[i][j+1]$)
- 2 which means wind in the cell flows to the left. (i.e from $S[i][j]$ to $S[i][j-1]$)
- 3 which means wind in the cell flows downwards. (i.e from $S[i][j]$ to $S[i+1][j]$)
- 4 which means wind in the cell flows upwards. (i.e from $S[i][j]$ to $S[i-1][j]$)

Notice that there could be some signs on the cells of the grid S that point outside the Grand Line sea grid.

- Luffy's ship *"Merry"* can only sail along the wind direction and can't go outside the Grand Line sea grid S at any point.
- Luffy can also modify the wind's direction on a cell with $cost = 1$. *(can modify the sign on a cell one time only)*

Find the minimum cost to make Luffy's Voyage from the top left corner of the Grand line i.e $S[1][1]$ to its bottom right corner i.e $S[N][M]$ possible.

Input Format

Input is given from Standard Input in the following format:

$N \ \ M$
$S_{1,1} \ldots S_{1,M}$
:
:
$S_{N,1} \ldots S_{N,M}$

Output Format

Print the answer.

Constraints

$2 \leq N, M \leq 1000$

C++14 | 00:00:00 | 12 px

```
 8  vector<vector<int>>g;
 9  vector<vector<int>>dis;
10  int dx[]={0,0,1,-1};
11  int dy[]={1,-1,0,0};
12  bool check(int x,int y){
13      if(x>=1 && x<=n && y>=1 && y<=m){
14          return true;
15      }
16      return false;
17  }
18  map<pp,int>neighbour(pp node){
19      map<pp,int>ans;
20      for(int i=0;i<4;i++){
21          int x=node.first+dx[i];
22          int y=node.second+dy[i];
23          if(check(x,y)){
24              if(g[node.first][node.second]==1 && x==node.
                first && y==node.second+1){
25                  ans[{x,y}]=0;
26              }
27              else if(g[node.first][node.second]==2 && x==node.
                first && y==node.second-1){
28                  ans[{x,y}]=0;
```

Sample Tests | Manual Tests

Test Case 1 | Test Case 2

ACCEPTED

Input

4 4

Console | Run on Sample