

16) LONGEST POSSIBLE ROUTE IN THE GRID CONTAINING HURDLES

METHOD:

This sum is the same as that of the rat in the maze problem. But here everytime we reach the destination we compare the number of steps with the maximum of the length previously achieved.

CODE OF THE PROGRAM:

```
#include<iostream>
#include<vector>
#include<limits.h>
using namespace std;

int length=INT_MIN;

void maxLength(vector<vector<int>> & grid,int r,int c,int currr,int currc,int steps,vector<int> & coordinates){
    if(currr==coordinates[2] && currc==coordinates[3]){
        length=max(length,steps);
        return;
    }
    else{
        int a[4]={-1,0,1,0};
        int b[4]={0,1,0,-1};
        for(int i=0;i<4;i++){
            if(currr+a[i]>=0 && currr+a[i]<r && currc+b[i]>=0 && currc+b[i]<c && grid[currr+a[i]][currc+b[i]]==1){
                int t=grid[currr][currc];
                grid[currr][currc]=2;
                maxLength(grid,r,c,currr+a[i],currc+b[i],steps+1,coordinates);
                grid[currr][currc]=t;
            }
        }
        return;
    }
}

int fun(vector<vector<int>> &grid,int r,int c,vector<int> & coordinates){
    maxLength(grid,r,c,coordinates[0],coordinates[1],0,coordinates);
    return length;
}

int main(){
    int r,c;
    cout<<"\n Enter the number of rows in the matrix:";
    cin>>r;
    cout<<"\n Enter the number of columns in the matrix:";
    cin>>c;
```

```

vector<vector<int>> grid(r,vector<int>(c));
cout<<"\n Enter the grid:";
for(int i=0;i<r;i++){
    for(int j=0;j<c;j++){
        cin>>grid[i][j];
    }
}
vector<int> coordinates(4);
cout<<"\n Enter the starting coordinates:";
cin>>coordinates[0]>>coordinates[1];
cout<<"\n Enter the ending coordinates:";
cin>>coordinates[2]>>coordinates[3];
int length=fun(grid,r,c,coordinates);
cout<<"\n The maximum distance between the starting and the ending coordinates:"<<length;
return 0;
}

/*

1 1 1 1 1 1 1 1 1
1 1 0 1 1 0 1 1 0 1
1 1 1 1 1 1 1 1 1 1

*/

```