5 4

0 0 1 0

0 0 0 0

0 0 1 0

0 1 0 0

0 0 0 1

1 1 4 3

//maze

// DFS

#include<bits/stdc++.h>

using namespace std;

const int N=101;//max size of maze

int maze[N][N]={0};

int record[N][N]={0};// 记录行走的过程

int dir[4][2]={{0,1},{1,0},{0,-1},{-1,0}}; //顺时针方向

int n,m;//实际行 和 列

int sx,sy,ex,ey;//起点，终点

int steps=99999;//设置最小步数

void dfs(int x,int y,int step)

{

int i,cx,cy;

if(x==ex && y==ey) // 找到目标点

{

if(step<steps) steps=step;

return ;

}

for(i=0;i<=3;i++)

{

//四个方向的点尝试

cx=x+dir[i][0];

cy=y+dir[i][1];

// 判断是否越界

if(cx<1||cx>n||cy<1||cy>m) continue;

// 判断是否有障碍 并且未走过

if(maze[cx][cy]==0&&record[cx][cy]==0)

{

record[cx][cy]=1;//标记

dfs(cx,cy,step+1);

record[cx][cy]=0;//取消标记

}

}

return ;

}

int main()

{

freopen("maze.in","r",stdin);

// freopen("maze.out","w",stdout);

int i,j;

cin>>n>>m;

for(i=1;i<=n;i++)

for(j=1;j<=m;j++)

cin>>maze[i][j];

cin>>sx>>sy>>ex>>ey;

record[sx][sy]=1;//设置起点记录

dfs(sx,sy,0);

cout<<"steps= "<<steps<<endl;

return 0;

}

5 4

0 0 1 0

0 0 0 0

0 0 1 0

0 1 0 0

0 0 0 1

1 1 4 3

//maze

// BFS

#include<bits/stdc++.h>

using namespace std;

struct node

{

int x,y;//坐标

int steps;//步数

};

const int N=101;//max size of maze

int maze[N][N]={0};

int record[N][N]={0};// 记录行走的过程

int dir[4][2]={{0,1},{1,0},{0,-1},{-1,0}}; //顺时针方向

int n,m;//实际行 和 列

int sx,sy,ex,ey;//起点，终点

int flag=0;//是否找到标志

int steps=99999;//设置最小步数

queue<node \*> q;// 遍历队列q;

void bfs()

{

int i,cx,cy;

node \*front,\*t,\*back;

while(!q.empty())

{

front=q.front();

for(i=0;i<4;i++)

{

//四个方向的点尝试

cx=front->x+dir[i][0];

cy=front->y+dir[i][1];

// 判断是否越界

if(cx<1||cx>n||cy<1||cy>m) continue;

// 判断是否有障碍 并且未走过

if(maze[cx][cy]==0&&record[cx][cy]==0)

{

record[cx][cy]=1;//标记

t=new node();

t->x=cx;t->y=cy;t->steps=front->steps+1;

q.push(t);

}

if(cx==ex &&cy==ey)

{

flag=1;//找到目标

back=q.back();

if(back->steps<steps) steps=back->steps;

break;

}

}

if(flag==1) break;

q.pop();

}

}

int main()

{

freopen("maze.in","r",stdin);

// freopen("maze.out","w",stdout);

int i,j;

cin>>n>>m;

for(i=1;i<=n;i++)

for(j=1;j<=m;j++)

cin>>maze[i][j];

cin>>sx>>sy>>ex>>ey;

struct node \*t=new node();

t->x=sx; t->y=sy; t->steps=0;

record[sx][sy]=1;//设置起点记录

q.push(t);

bfs();

cout<<"steps= "<<steps<<endl;

return 0;

}