

Answer  
  
public int IslandCount(char [] [] grid){

int ans=0;

for(int i=0;i<grid.length;i++)

{

for(int j=0;j<grid[1].length;j++)

{

if(grid[i][j]=='1')

{

ans+=1;

BFS(grid,i,j);

}

}

}

}

public void BFS (char [] [] grid,int i , int j)

{

if(i<0 || i>=A.length|| j<0 || jj>=grid[i].length||A[i][j]==0) return;

grid[i][j]='0';

// for up

BFS(grid,i+1,j);

BFS(grid,i-1,j);doe=wn

BFS(grid,i,j+1); //left

BFS(grid,i,j-1); //right

}

}

Question 2  
  
public class MinTimeToVisitPoints {

public static int minTimeToVisitAllPoints(int[][] points) {

int totalTime = 0;

for (int i = 1; i < points.length; i++) {

int x1 = points[i - 1][0]; // previous x

int y1 = points[i - 1][1]; // previous y

int x2 = points[i][0]; // current x

int y2 = points[i][1]; // current

int dx = Math.abs(x2 - x1);//currecnt x-previous one (3-1)=2

int dy = Math.abs(y2 - y1); // vertical steps(3-2)=1

int time = Math.max(dx, dy);

totalTime += time; // Add time for this move to total

}

return totalTime; }