

一、编程题

ACM:反射计数

题目描述:

给定一个包含 0 和 1 的二维矩阵

给定一个初始位置和速度

一个物体从给定的初始位置触发,在给定的速度下进行移动,遇到矩阵的边缘则发生镜面反射

无论物体经过 0 还是 1 , 都不影响其速度

请计算并给出经过 t 时间单位后, 物体经过 1 点的次数

矩阵以左上角位置为 $[0, 0]$ (列(x), 行(y)), 例如下面 A 点坐标为 $[2, 1]$ (第二列, 第一行)

+----- 递增(x)

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

| 0 0 1 0 0 0 0 1 0 0 0 0

|

递增(y)

注意:

如果初始位置的点是 1 , 也计算在内

时间的最小单位为 1 ，不考虑小于 1 个时间单位内经过的点

输入描述：

第一行为初始信息

$\langle w \rangle \langle h \rangle \langle x \rangle \langle y \rangle \langle sx \rangle \langle sy \rangle \langle t \rangle$

第二行开始一共 h 行，为二维矩阵信息

其中

w, h 为矩阵的宽和高

x, y 为起始位置

sx, sy 为初始速度

t 为经过的时间

所有输入都是有效的，数据范围如下

$$0 < w < 100$$

$$0 < h < 100$$

$$0 \leq x < w$$

$$0 \leq y < w$$

$$-1 \leq sx \leq 1$$

$$-1 \leq sy \leq 1$$

$0 \leq t < 100$

输出描述:

经过 1 的个数

注意初始位置也要计算在内

补充说明:

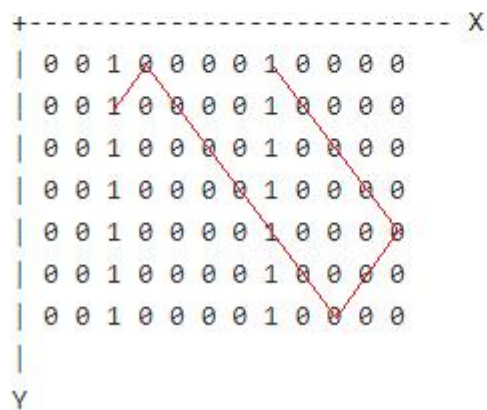
示例1

输入: 12 7 2 1 1 -1 13

```
001000010000
001000010000
001000010000
001000010000
001000010000
001000010000
001000010000
001000010000
```

输出: 3

说明: 初始位置为(2, 1), 速度为(1, -1), 那么13个时间单位后, 经过点1的个数为3



代码:

```
import java.util.Scanner;
```

// 注意类名必须为 Main, 不要有任何 package xxx 信息

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        Scanner in = new Scanner(System.in);
```

```
        // 注意 hasNext 和 hasNextLine 的区别
```

```

while (in.hasNext()) { // 注意 while 处理多个 case
    int w = in.nextInt();
    int h = in.nextInt();
    int x = in.nextInt();
    int y = in.nextInt();
    int sx = in.nextInt();
    int sy = in.nextInt();
    int t = in.nextInt();

    int[][] arrays = new int[h][w];

    for (int i = 0; i < h; i++) {
        String s = in.next();
        for (int j = 0; j < w; j++) {
            arrays[i][j] = Integer.parseInt(String.valueOf(s.charAt(j)));
        }
    }

    int sum = 0;

    if (w == 1 && h == 1) {
        sum = t * arrays[0][0];
        System.out.println(sum);
        return;
    }
    for (int i = 0; i <= t; i++) {
        if (x == 0 || x == w - 1) {
            if (i == 0) {
                if (x == 0) {
                    sx = 1;
                } else if (x == w - 1) {
                    sx = -1;
                }
            } else {
                sx = -sx;
            }
        }
        if (y == 0 || y == h - 1) {
            if (i == 0) {
                if (y == 0) {
                    sy = 1;
                } else if (y == h - 1) {

```

```

        sy = -1;
    }
    } else {
        sy = -sy;
    }
}
sum += arrays[y][x];
if (w > 1) {
    x += sx;
}
if (h > 1) {
    y += sy;
}
}
System.out.println(sum);
}
}
}

```