

信号发射和接收

知识点数组栈 单调栈 🔍

时间限制：1s 空间限制：256MB 限定语言：不限

题目描述：

有一个二维的 **天线** 矩阵，每根天线可以向其他天线发射信号也能接收其他天线的信号，为了简化起见，我们约定每根天线只能向东和向南发射信号，换言之，每根天线只能接收东向或南向发送的信号。

每根天线有自己的高度 $anth$ ，各根天线的高度存储在一个二维数组中，各个天线的位置用 $[r, c]$ 表示， r 代表天线的行位置（从0开始编号）， c 代表天线的列位置（从0开始编号）

在某一个方向（东向或南向），某根天线可以收到多根其他天线的信号（也可能收不到任何其他天线的信号），对任一天线 X 和天线 Y ，天线 X 能接收到天线 Y 的信号的条件是：（1）天线 X 在天线 Y 的东边或南边；（2）天线 X 和天线 Y 之间的其他天线的高度都低于天线 X 和天线 Y ，或天线 X 和天线 Y 之间无其他天线，即无遮挡。

如下图1示意：在天线矩阵的第0行上，天线 $[0, 0]$ 接收不到任何其他天线的信号，天线 $[0, 1]$ 可以接收到天线 $[0, 0]$ 的信号，天线 $[0, 2]$ 可以接收到天线 $[0, 1]$ 的信号，天线 $[0, 3]$ 可以接收到天线 $[0, 1]$ 和天线 $[0, 2]$ 的信号，天线 $[0, 4]$ 可以接收到天线 $[0, 3]$ 的信号，天线 $[0, 5]$ 可以接收到天线 $[0, 4]$ 的信号；在天线矩阵的第0列上，天线 $[0, 0]$ 接收不到任何其他天线的信号，天线 $[1, 0]$ 可以接收到天线 $[0, 0]$ 的信号，天线 $[2, 0]$ 可以接收到天线 $[1, 0]$ 的信号，天线 $[3, 0]$ 可以接收到天线 $[1, 0]$ 和天线 $[2, 0]$ 的信号，天线 $[4, 0]$ 可以接收到天线 $[3, 0]$ 的信号，天线 $[5, 0]$ 可以接收到天线 $[3, 0]$ 和天线 $[4, 0]$ 的信号

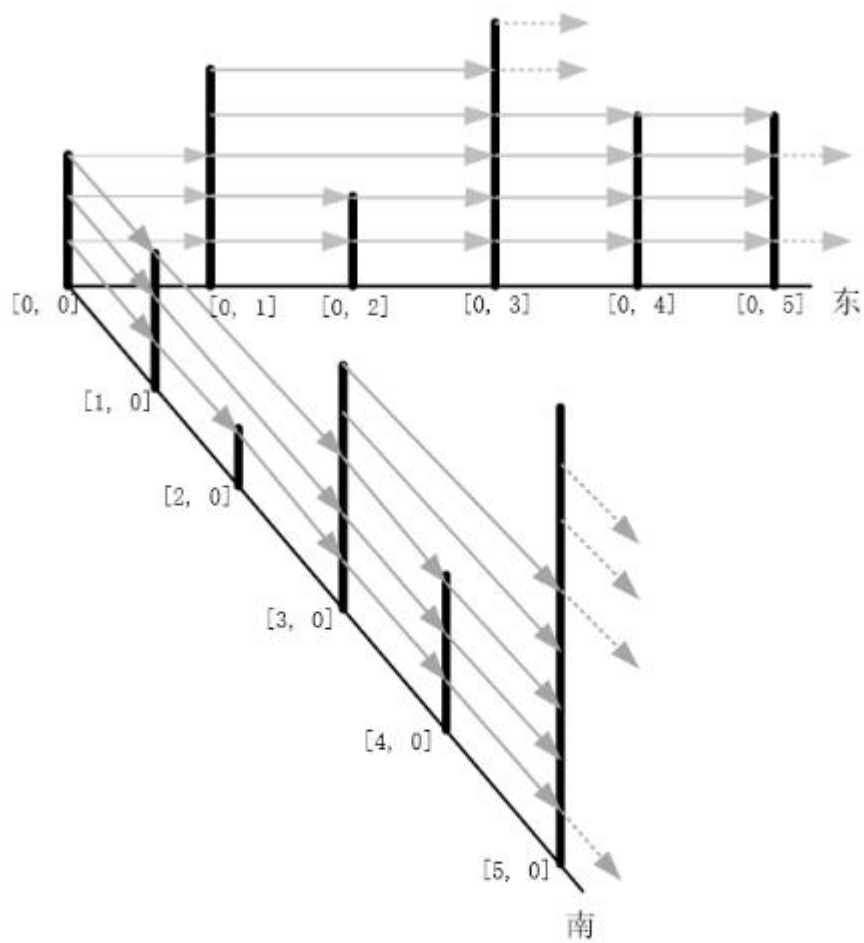


图1

给一个m行n列的矩阵（二维数组），矩阵存储各根天线的高度，求出每根天线可以收到多少根其他天线的信号，结果输出到m行n列的矩阵（二维数组）中。

输入描述:

输入为1个m行n列的矩阵（二维数组）`anthh[m][n]`，矩阵存储各根天线的高度，高度值`anthh[r][c]`为大于0的整数。具体示例如下：

m n

`anth[0][0] anth[0][1] ... anth[0][n-1] anth[1][0] anth[1][1] ... anth[1][n-1] ... anth[m-1][0] ... anth[m-1][n-1]`

第1行为输入矩阵的行数和列数

第2行为输入矩阵的元素值，按行输入

输出描述:

输出1个m行n列的矩阵（二维数组）`ret[m][n]`，矩阵存储每根天线能收到多少根其他天线的信号，根数为`ret[r][c]`。具体示例如下：

m n

`ret[0][0] ret[0][1] ... ret[0][n-1] ret[1][0] ret[1][1] ... ret[1][n-1] ... ret[m-1][0] ... ret[m-1][n-1]`

第1行为输出矩阵的行数和列数

第2行为输出矩阵的元素值，按行输出

补充说明:

$1 \leq m \leq 500$

$1 \leq n \leq 500$

$0 < \text{ant}[r][c] < 10^5$

示例1

输入：

```
1 6
2 4 1 5 3 3
```

输出：

```
1 6
0 1 1 2 1 1
```

说明：

输入为1行6列的天线矩阵的高度值

```
[2 4 1 5 3 3]
```

输出为1行6列的结果矩阵

```
[0 1 1 2 1 1]
```

示例2

输入：

```
2 6
2 5 4 3 2 8 9 7 5 10 10 3
```

输出：

```
2 6
0 1 1 1 1 4 1 2 2 4 2 2
```

说明：

输入为2行6列的天线矩阵高度值

```
[2 5 4 3 2 8]
```

```
[9 7 5 10 10 3]
```

输出为2行6列的结果矩阵

```
[0 1 1 1 1 4]
```

```
[1 2 2 4 2 2]
```

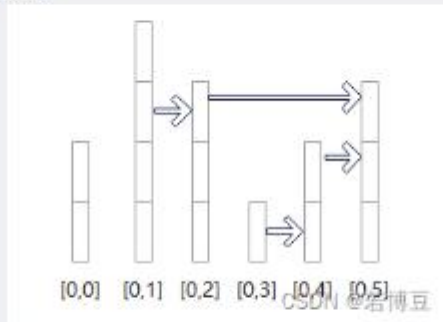
结果说明：

天线[0,0]收不到任何其他天线的信号，因此ret[0,0]=0；天线[0, 1]可接收到天线[0, 0]的信号，因此ret[0, 1]=1；天线[0, 2]可以接收到天线[0, 1]的信号，因此ret[0, 2]=1；天线[0, 3]可以接收到天线[0, 2]的信号，因此ret[0, 3]=1；天线[0, 4]可以接收到天线[0, 3]的信号，因此ret[0, 4]=1；天线[0, 5]可以接收到天线[0, 1]、天线[0, 2]、天线[0, 3]、天线[0, 4]的信号，因此ret[0, 5]=4

天线[1, 0]可以接收到天线[0, 0]的信号，因此ret[1, 0]=1；天线[1, 1]可以接收到天线[0, 1]、天线[1, 0]的信号，因此ret[1, 1]=2；天线[1, 2]可以接收到天线[0, 2]、天线[1, 1]的信号，因此ret[1, 2]=2；天线[1, 3]可以接收到天线[0, 3]、天线[1, 0]、天线[1, 1]、天线[1, 2]的信号，因此ret[1, 3]=4；天线[1, 4]可以接收到天线[0, 4]、天线[1, 3]的信号，因此ret[1, 4]=2；天线[1, 5]可以接收到天线[0, 5]、天线[1, 4]的信号，因此ret[1, 5]=2

解题思路:

求天线接收信号的个数，如图：



如求[0,5]可接收的信号：

以左侧第一根天线为最大值 $\max = [0,4] = 2$ ，可以接收到信号， $\text{count}=1$ ；

$[0,3]=1 < \max$ ，接收不到此天线的信号， $\text{count}=1$ ；

$[0,2]=3 == [0,5] > \max$ ，可以接收到信号， $\text{count}=2$ ，且 $\max = 3$ ；因为与接收天线等高，所以往后的天线都没有办法接收到信号了。所以 $\text{count}=2$ 。

```
public class Main{

    public static int[][] ints;

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int r = sc.nextInt();
        int c = sc.nextInt();
        ints = new int[r][c];

        for(int i=0;i<r;i++){
            for(int j=0;j<c;j++){
                ints[i][j] = sc.nextInt();
            }
        }

        String res = "";
        for(int i=0;i<r;i++){
            for(int j=0;j<c;j++){
                int count = rowSignal(i,j) + colSignal(i,j);
                res += count + " ";
            }
        }
    }
}
```

```

        System.out.println(r + " " + c);
        System.out.println(res.substring(0,res.length()-1));

    }

    /**
     * 求行接收信号数
     * @param row
     * @param col
     * @return
     */
    public static int rowSignal(int row, int col){

        if(col==0){        //在第 1 列没有行信号接收
            return 0;
        }
        int max = ints[row][col-1];        //以此天线左侧的第一根天线作为最大值
        int count = 1; //左侧第一根一定能接收到信号，所以至少有 1 个

        for(int i=col-2; i>=0; i--){
            if(max >= ints[row][col]){ //当最大高度大于等于接收天线时，后面的天线发射
的信号都无法接收
                break;
            }
            int height = ints[row][i];
            if(height>max){        //此地的天线是当前最高天线时，表示可以接收到信号
                count++;
                max = height;
            }
        }

        return count;
    }

    /**
     * 求列信号接收数
     * @param row
     * @param col
     * @return
     */
    public static int colSignal(int row, int col){

        if(row==0){        //在第 1 行没有列信号接收

```

```

        return 0;
    }
    int max = ints[row-1][col];    //以此天线上面的第一根天线为最大值
    int count = 1;                //上侧第一根一定能接收到信号，所以至少有 1 个

    for(int i=row-2; i>=0; i--){
        if(max >= ints[row][col]){ //当最大高度大于等于接收天线时，后面的天线发射
的信号都无法接收
            break;
        }
        int height = ints[i][col];
        if(height>max){           //此地的天线是当前最高天线时，表示可以接收到信号
            count++;
            max = height;
        }
    }

    return count;
}
}

```