

Imagen de Unique Path II Ejercicio1

Java

Autocomplete

```
1 class Solution {
2     public int uniquePathsWithObstacles(int[][] obstacleGrid) {
3         int m = obstacleGrid.length;
4         int n = obstacleGrid[0].length;
5         int temp[] = new int[n];
6         if (obstacleGrid[0][0] == 1)
7             return 0;
8         temp[0] = 1;
9
10        for (int i = 1; i < n; i++) {
11            temp[i] = 0;
12            if (temp[i-1] == 1 && obstacleGrid[0][i] == 0) {
13                temp[i] = 1;
14            }
15        }
16        return temp[n-1];
17    }
18 }
```

Your previous code was restored from your local storage. [Reset to default](#)

Testcase

Run Code Result

Debugger

Accepted

Runtime: 0 ms

Your input

[[0,0,0],[0,1,0],[0,0,0]]
[[0,1],[0,0]]

Output

2
1

Expected

2
1

Console

Use Example Testcases

?

Run

Imagen Number Long Ejercicio 3

re

☆ Premium



```
Java Autocomplete i {}  
  
1 class Solution {  
2     public int lengthOfLIS(int[] nums) {  
3         int []aux=new int[nums.length]; //aux toma valores maximos de cadena creciente  
4         aux[0]=1;  
5         for(int i=1;i<nums.length;i++){  
6             int cont=1;  
7             for(int j=0;j<i;j++){  
8                 //compara valores crecientes, si es menor aumentara contador,  
9                 //y luego actualiza el auxiliar con el contador  
10                if(nums[j]<nums[i]){  
11                    cont=Math.max(cont,aux[j]+1);  
12                }  
13                aux[i]=cont;  
14            }  
15        }  
16        return aux[nums.length-1];  
17    }  
18 }
```

testcase Run Code Result Debugger

Accepted Runtime: 0 ms

Your input	[10,9,2,5,3,7,101,18] [0,1,0,3,2,3] [7,7,7,7,7,7,7]
Output	4 4 1
Expected	4 4 1

Console Use Fullscreen Testcase

Run Code

Imagen Maximal Square Ejercicio 5

Store

LeetCode Challenge + GIVEAWAY!

Premium

Java Autocomplete

i {}

```
1 class Solution {
2     public int maximalSquare(char[][] matrix) {
3         int n,m;
4         n=matrix.length;
5         m=matrix[0].length;
6         int[][] B=new int[n][m];
7         for(int i=0;i<n;i++) {
8             if(matrix[i][0]=='1')
9                 B[i][0]=1;
10            else
11                B[i][0]=0;
12        }
13        for(int j=0;j<m;j++) {
14            if(matrix[0][j]=='1')
```

Your previous code was restored from your local storage. [Reset to default](#)

Testcase Run Code Result Debugger

Accepted Runtime: 0 ms

Your input

```
[[["1","0","1","0","0"],["1","0","1","1","1"],["1","1","1","1","1"],["1","0","0","1","0"]],
[["0","1"],["1","0"]],
[["0"]]]
```

Output

```
4
1
0
```

Expected

```
4
1
0
```

Console Use Example Testcases

Run Code