

Laboratorio Atestado de Perros*Contribución de Lautaro Lasorsa***Descripción del problema**

Durante el Training Camp Argentina 2019, realizado en Córdoba, fue necesario utilizar el laboratorio de informática para realizar competencias. El problema es que en la UNC hay muchos perros dando vueltas, y los perros fueron entrando al laboratorio y obstruyendo el paso.

Más específicamente, el laboratorio puede representarse como una grilla de $N \times M$ baldosas. Una baldosa puede estar libre, es decir se puede transitar sobre ella, o puede haber mobiliario (escritorios para computadoras principalmente) sobre ella, que no puede ser movido e impide transitarla.

Es importante marcar que el laboratorio tiene una única entrada, pero que puede abarcar varias baldosas.

Se dice que una baldosa libre es accesible si es posible armar un camino que pase solo por baldosas libres, pudiendo moverse desde una baldosa solo hacia arriba, abajo, izquierda y derecha, que una la entrada con dicha baldosa. Normalmente (cuando no hay perros), todas las baldosas libres son accesibles.

Además, han ingresado, todos en momentos distintos, P perros, de la siguiente forma :

- Cuando un perro entra, se recuesta en una baldosa libre, bloqueandola (no es posible transitar sobre una baldosa en la que hay un perro). Como está muy cómodo ahí permanecerá indefinidamente en ese lugar.
- Los perros son ágiles y además se dejan pasar entre ellos, por lo que un perro podrá recostarse en una baldosa incluso si esta no es accesible para las personas.

Dados los lugares en los que se recosto

cada perro, en el orden en que fueron entrando, se desea saber cuántas baldosas accesibles había después de la entrada de cada perro.

Detalles de implementación

Se debe implementar la función `laboratorio(labo,perros)`

- `labo` es un vector de largo N de cadenas de caracteres de largo M , donde la posición `labo[i][j]` describe la baldosa en fila i y columna j . Su valor será :
 - '.' si es una baldosa libre.
 - '#' si es una baldosa con mobiliario.
 - 'E' si es una baldosa adyacente a la entrada. Todas las baldosas 'E' están en el borde del laboratorio, y son adyacentes entre sí. Las baldosas 'E' son, a su vez, baldosas libres.
- `perros` es un vector de largo P de pares de enteros, donde el par `perros[i]` indica en qué baldosa se ubicará el i –ésimo perro.

La función debe retornar un vector de enteros de largo P , donde la i –ésima posición contenga la cantidad de baldosas libres accesibles después de que entre el i –ésimo perro.

Evaluador local

El evaluador local lee de la entrada estándar:

- Una línea con los valores N , M y P
- N líneas de M caracteres cada una, sin espacios, describiendo el estado inicial del laboratorio. (el contenido de labo)
- Luego P líneas, cada una con 2 enteros. El primero entre 0 y $N - 1$, y el segundo entre 0 y $M - 1$, son los valores de `perros[i]`

Devuelve por la salida estándar una línea con los valores del vector retornado por laboratorio.

Cotas

- $1 \leq N \times M \leq 1.000.000$
- $0 \leq P \leq 100.000$

Ejemplos

Si el evaluador local recibe la siguiente entrada:

```
4 4 5
...#
#...
E..#
E..#
1 3
0 0
0 1
2 0
3 0
```

Una implementación correcta deberá devolver:

```
11 10 9 8 0
```

Si en cambio recibe:

```
1 10 5
#...E...#
0 2
0 1
0 5
0 6
0 7
```

Una implementación correcta devolverá:

```
6 6 0 0 0
```

Y si en cambio recibe :

```
10 10 6
E..#...#..
E..#...#..
E..#...#..
E..###.###
E.....
E..###.###
E..#...#..
E..#...#..
E..#...#..
E..#...#..
3 5
5 5
3 8
5 8
4 7
4 3
```

Devolverá:

```
66 53 46 37 34 30
```

Subtareas

1. $P = 1$ (10 puntos)
2. $N \times M \times P \leq 1.000.000$ (20 puntos)
3. $N = 1$ (20 puntos)
4. Sin restricciones adicionales (50 puntos)