

Encontrando Goteras

Contribución de Facundo Martín Gutiérrez

Descripción del problema

Frustrado por no haber podido organizar su depósito, Nicolás vendió todas las baratijas que en él tenía guardadas utilizando el conocido sitio de ventas en línea *Mercachifle.com*.

El depósito puede modelarse como un largo pasillo, separado en baldosas numeradas desde **1** hasta **L** inclusive. Ya no quedan baratijas, pero por encima de algunas baldosas existen goteras. Nicolás sabe que hay exactamente **N** baldosas con goteras, y que **todas las baldosas con gotera están numeradas con un número par entre 1 y L inclusive**.

Como el depósito está muy oscuro, no puede ver dónde están las goteras. Sin embargo, puede tratar de localizarlas utilizando un artilugio especial: una especie de “sensor con pinzas expansibles” que compró en *Mercachifle.com* hace mucho tiempo.

El sensor se utiliza de la siguiente manera:

- Se elige un número de baldosa **x**. $1 \leq x \leq L$.
- El artilugio rápidamente expande sus pinzas, realiza diversas mediciones, y arroja como resultado un único entero **d**: La distancia, medida en baldosas, desde la baldosa **x** hasta la baldosa más cercana con goteras.

Este artilugio consume para su operación un combustible exótico y extremadamente caro. Es por esto que Nicolás desea realizar la menor cantidad de mediciones posible para encontrar las goteras.

Debes ayudar a Nicolás escribiendo una función que, realizando las mediciones que sean necesarias, determine las ubicaciones exactas de las **N** baldosas con goteras.

Descripción de la función

Debes implementar la función `goteras(N, L)`, que recibe:

- **N**: Un entero con la cantidad total de goteras.
- **L**: Un entero con la cantidad total de baldosas.

La función retorna un arreglo de longitud **N**, con los números de las baldosas que contienen goteras, en cualquier orden.

Además, la implementación de `goteras` puede llamar a la función `medir(x)`, que retorna la distancia **d** desde la baldosa **x** hasta aquella más cercana a **x** con goteras. **Debe ser $1 \leq x \leq L$** .

El puntaje de su solución dependerá de la cantidad de veces que se llame la función `medir` (ver Puntuación).

Evaluador

El evaluador local lee de la entrada estándar con el siguiente formato:

- Una línea con dos enteros **N** y **L**
- Una línea con **N** enteros que indican las ubicaciones de las goteras.

El evaluador local escribe a la salida estándar una línea por cada llamada `medir(x)` realizada, indicando además la respuesta. Finalmente, escribe una línea con el resultado de la función.

Cotas

- $1 \leq N \leq 1.000$
- $2N \leq L \leq 10^{18}$
- Todas las baldosas con gotera tienen número par entre **1** y **L** inclusive.
- En toda llamada a `medir(x)`, debe ser obligatoriamente $1 \leq x \leq L$

Ejemplo

Si se invoca al evaluador con la siguiente entrada:

```
5 40
2 22 20 30 38
```

La siguiente es una posible salida:

```
medir(14) = 6
medir(20) = 0
medir(5) = 3
medir(26) = 4
medir(40) = 2
medir(30) = 0
medir(22) = 0
medir(2) = 0
2 20 22 38 30
```

En este ejemplo, el programa habría realizado las 8 mediciones indicadas, y finalmente habría retornado correctamente las ubicaciones de las goteras.

Evaluador adaptativo

El evaluador para este problema es adaptativo. Eso significa que las baldosas con goteras pueden no estar prefijadas, sino depender de las mediciones realizadas por el programa.

Esto no afecta a un programa correcto que funcione siempre para cualquier entrada posible, ya que las respuestas del evaluador serán en todo momento consistentes con alguna elección de baldosas.

Puntuación y subtareas

Si el programa no determina correctamente la ubicación de las baldosas en **todos** los casos de prueba de una subtarea, el puntaje final para esa subtarea será 0.

Si el programa determina correctamente las ubicaciones, el puntaje obtenido en la subtarea dependerá de la cantidad máxima **P** de preguntas que se haya realizado en algún caso de prueba de esa subtarea, de acuerdo a la siguiente tabla:

1. **N = 1** (2 puntos máximo):
 - **P ≤ 1**: 2 puntos
 - **2 ≤ P ≤ 1.000**: 1 punto
2. **N = 2** (5 puntos máximo):
 - **P ≤ 2**: 5 puntos
 - **P = 3**: 2 puntos
 - **4 ≤ P ≤ 1.000**: 1 punto
3. **N = 3** (18 puntos máximo):
 - **P ≤ 4**: 18 puntos
 - **P = 5**: 10 puntos
 - **P = 6**: 7 puntos
 - **P = 7**: 3 puntos
 - **8 ≤ P ≤ 1.000**: 2 puntos
4. **4 ≤ N ≤ 1.000** (75 puntos máximo):
 - **P ≤ 2995**: 75 puntos
 - **P = 2996**: 72 puntos
 - **P = 2997**: 70 puntos
 - **P = 2998**: 68 puntos
 - **P = 2999**: 67 puntos
 - **3000 ≤ P ≤ 6000**: 36 puntos
 - **6001 ≤ P ≤ 11000**: 15 puntos
 - **11000 < P**: 8 puntos