

Comparando Plantas (plants)

Hazel acaba de visitar una exhibición especial en los Jardines Botánicos de Singapur. En esta exhibición, n plantas de **diferentes alturas** están ubicadas en un círculo. Estas plantas están numeradas de 0 a $n - 1$ en sentido horario. La planta $n - 1$ está al lado de la planta 0.

Para cada planta i ($0 \leq i \leq n - 1$), Hazel comparó la planta i con cada una de las siguientes $k - 1$ plantas en sentido horario y anotó el número $r[i]$ que indica cuántas de estas $k - 1$ plantas son más altas que la planta i . Es decir, cada valor $r[i]$ depende de las alturas relativas de algunas de las k plantas consecutivas.

Por ejemplo, supongamos que $n = 5$, $k = 3$ y $i = 3$. Las siguientes $k - 1 = 2$ plantas en sentido horario desde la planta $i = 3$ serían la planta 4 y la planta 0. Si la planta 4 fuera más alta que la planta 3 y la planta 0 fuera más baja que la planta 3, Hazel anotaría $r[3] = 1$.

Puedes asumir que Hazel anotó los valores $r[i]$ correctamente. Es decir, existe al menos una configuración de alturas distintas que es consistente con estos valores.

Te pidieron comparar las alturas de q pares de plantas. Desafortunadamente, no tienes acceso a la exhibición. Tu única fuente de información es el cuaderno de Hazel con el valor k y la secuencia de valores $r[0], \dots, r[n - 1]$.

Para cada par de plantas x y y a comparar, debes determinar cuál de las tres siguientes situaciones ocurre:

- La planta x es definitivamente más alta que la planta y : en cualquier configuración de alturas diferentes $h[0], \dots, h[n - 1]$ consistente con el arreglo r se tiene que $h[x] > h[y]$.
- La planta x es definitivamente más baja que la planta y : en cualquier configuración de alturas diferentes $h[0], \dots, h[n - 1]$ consistente con el arreglo r tenemos $h[x] < h[y]$.
- La comparación no tiene un resultado concluyente: ninguna de las dos situaciones previas aplica.

Detalles de implementación

Debes implementar las siguientes funciones:

```
void init(int k, int[] r)
```

- k : el número de plantas consecutivas cuyas alturas determinan cada valor de $r[i]$.
- r : un arreglo de tamaño n , donde $r[i]$ es el número de plantas más altas que la planta i , dentro de las siguientes $k - 1$ plantas en sentido horario.

- Esta función es llamada exactamente una vez, antes de cualquier llamada a `compare_plants`.

```
int compare_plants(int x, int y)
```

- x, y : números de las plantas a comparar.
- Esta función debe retornar:
 - 1 si la planta x es definitivamente más alta que la planta y ,
 - -1 si la planta x es definitivamente más baja que la planta y ,
 - 0 si la comparación no tiene un resultado concluyente.
- Esta función es llamada exactamente q veces.

Ejemplos

Ejemplo 1

Considera la siguiente llamada:

```
init(3, [0, 1, 1, 2])
```

Supongamos que el evaluador llama a `compare_plants(0, 2)`. Como $r[0] = 0$, podemos inferir inmediatamente que la planta 2 no es más alta que la planta 3. Es decir, la llamada debe retornar 1.

Supongamos ahora que el evaluador luego llama a `compare_plants(1, 2)`. Para todas las posibles configuraciones de alturas que cumplen con las restricciones anteriores, la planta 1 es más baja que la planta 2. Entonces, la llamada debe retornar -1 .

Ejemplo 2

Considera la siguiente llamada:

```
init(2, [0, 1, 0, 1])
```

Supongamos que el evaluador llama a `compare_plants(0, 3)`. Como $r[3] = 1$, sabemos que la planta 0 es más alta que la planta 3. Entonces, la llamada debe retornar 1.

Supongamos ahora que el evaluador luego llama a `compare_plants(1, 3)`. Las configuraciones de alturas $[3, 1, 4, 2]$ y $[3, 2, 4, 1]$ son ambas consistentes con las medidas de Hazel. Como la planta 1 es más baja que la planta 3 en una configuración y más alta que la planta 3 en otra, esta llamada debe retornar 0.

Restricciones

- $2 \leq k \leq n \leq 200\,000$

- $1 \leq q \leq 200\,000$
- $0 \leq r[i] \leq k - 1$ (para todo $0 \leq i \leq n - 1$)
- $0 \leq x < y \leq n - 1$
- Existen una o más configuraciones de **distintas alturas** de plantas consistentes con el arreglo r .

Subtareas

1. (5 puntos) $k = 2$
2. (14 puntos) $n \leq 5000, 2 \cdot k > n$
3. (13 puntos) $2 \cdot k > n$
4. (17 puntos) La respuesta correcta a cada llamada de `compare_plants` es 1 o -1 .
5. (11 puntos) $n \leq 300, q \leq \frac{n \cdot (n-1)}{2}$
6. (15 puntos) $x = 0$ para cada llamada a `compare_plants`.
7. (25 puntos) No hay restricciones adicionales.

Evaluador de prueba

El evaluador de prueba lee la entrada en el siguiente formato:

- línea 1: $n \ k \ q$
- línea 2: $r[0] \ r[1] \ \dots \ r[n - 1]$
- línea $3 + i$ ($0 \leq i \leq q - 1$): $x \ y$ para la i -ésima llamada a `compare_plants`

El evaluador de prueba imprime tus respuestas en el siguiente formato:

- línea $1 + i$ ($0 \leq i \leq q - 1$): el valor que regresó la i -ésima llamada a `compare_plants`.