

Comparando Plantas (plants)

A botanista Hazel visitou uma exposição especial no Jardim Botânico de Singapura. Nessa exposição, n plantas de **alturas distintas** são colocadas em um círculo. Essas plantas são rotuladas de 0 até $n - 1$ no sentido horário, com a planta $n - 1$ ao lado da planta 0.

Para cada planta i ($0 \leq i \leq n - 1$), Hazel comparou a planta i com cada uma das próximas $k - 1$ plantas no sentido horário, e anotou o número $r[i]$ denotando quantas dessas $k - 1$ plantas são mais altas do que a planta i . Assim, cada valor $r[i]$ depende das alturas relativas de algumas k plantas consecutivas.

Por exemplo, suponha que $n = 5$, $k = 3$ e $i = 3$. As próximas $k - 1 = 2$ plantas no sentido horário a partir da planta $i = 3$ seriam a planta 4 e a planta 0. Se a planta 4 fosse mais alta do que a planta 3 e a planta 0 fosse mais baixa do que a planta 3, Hazel teria anotado $r[3] = 1$.

Você pode assumir que Hazel registrou os valores $r[i]$ corretamente. Portanto, há pelo menos uma configuração de plantas de alturas diferentes consistente com esses valores.

Pediram que você compare as alturas de q pares de plantas. Infelizmente você não tem acesso à exposição. Sua única fonte de informação é o caderno de anotações de Hazel contendo o valor k e a sequência de valores $r[0], \dots, r[n - 1]$.

Para cada par de plantas diferentes x e y que necessitam ser comparadas, determine qual das seguintes três situações ocorre:

- A planta x é definitivamente mais alta do que a planta y : em qualquer configuração de alturas distintas $h[0], \dots, h[n - 1]$ consistente com o array r , temos $h[x] > h[y]$.
- A planta x é definitivamente mais baixa do que a planta y : em qualquer configuração de alturas distintas $h[0], \dots, h[n - 1]$ consistente com o array r , temos $h[x] < h[y]$.
- A comparação é inconclusiva: nenhum dos dois casos anteriores se aplica.

Detalhes de implementação

Você deve implementar os seguintes procedimentos:

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

- k : o número de plantas consecutivas cujas alturas determinam cada valor individual $r[i]$.
- r : um array de tamanho n , onde $r[i]$ é o número de plantas mais altas do que a planta i entre as próximas $k - 1$ plantas em sentido horário.

- Este procedimento é chamado exatamente uma vez, antes de qualquer chamada de `compare_plants`.

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

- x, y : rótulos das plantas que devem ser comparadas.
- Este procedimento deve retornar:
 - 1 se a planta x é definitivamente mais alta do que a planta y ,
 - -1 se a planta x é definitivamente mais baixa do que a planta y ,
 - 0 se a comparação é inconclusiva.
- Este procedimento é chamado exatamente q vezes.

Exemplos

Exemplo 1

Considere a seguinte chamada:

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

Digamos que o corretor faça a chamada `compare_plants(0, 2)`. Como $r[0] = 0$ podemos imediatamente inferir que a planta 2 não é maior do que a planta 0. Portanto, a chamada deve retornar 1.

Digamos que o corretor em seguida faça a chamada `compare_plants(1, 2)`. Para todas as possíveis configurações de alturas que obedecem às restrições acima, a planta 1 é mais baixa do que a planta 2. Portanto, a chamada deve retornar -1 .

Exemplo 2

Considere a seguinte chamada:

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

Digamos que o corretor faça a chamada `compare_plants(0, 3)`. Como $r[3] = 1$, sabemos que a planta 0 é mais alta do que a planta 3. Portanto, a chamada deve retornar 1.

Digamos que o corretor faça a chamada em seguida `compare_plants(1, 3)`. Duas configurações de alturas $[3, 1, 4, 2]$ e $[3, 2, 4, 1]$ são ambas consistentes com as anotações de Hazel. Como a planta 1 é mais baixa do que a planta 3 em uma configuração e mais alta do que a planta 3 na outra, a chamada deve retornar 0.

Restrições

- $2 \leq k \leq n \leq 200\,000$
- $1 \leq q \leq 200\,000$
- $0 \leq r[i] \leq k - 1$ (para todo $0 \leq i \leq n - 1$)
- $0 \leq x < y \leq n - 1$
- Existe uma ou mais configurações de **alturas distintas** de plantas consistente com o array r .

Subtarefas

1. (5 pontos) $k = 2$
2. (14 pontos) $n \leq 5000, 2 \cdot k > n$
3. (13 pontos) $2 \cdot k > n$
4. (17 pontos) A resposta correta para cada chamada de `compare_plants` é 1 ou -1 .
5. (11 pontos) $n \leq 300, q \leq \frac{n \cdot (n-1)}{2}$
6. (15 pontos) $x = 0$ para cada chamada de `compare_plants`.
7. (25 pontos) Nenhuma restrição adicional.

Corretor exemplo

O corretor exemplo lê a entrada no seguinte formato:

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

O corretor exemplo escreve suas respostas no seguinte formato:

- linha $1 + i$ ($0 \leq i \leq q - 1$): valor de retorno da i -ésima chamada de `compare_plants`.