

## Порівняння рослин (plants)

Ботанік Хазел потрапив на спеціальну виставку у Сінгапурському ботанічному саду. На цій виставці  $n$  рослин **різної висоти** розташовані по колу. Ці рослини пронумеровані від 0 до  $n - 1$  за годинниковою стрілкою, з рослиною  $n - 1$  поруч розташована рослина 0.

Для кожної рослини  $i$  ( $0 \leq i \leq n - 1$ ), Хазел порівняв рослину  $i$  з кожною з наступних  $k - 1$  рослин за годинниковою стрілкою та записав число  $r[i]$ , що означає кількість з цих  $k - 1$  рослин що вище рослини  $i$ . Отже, кожне з чисел  $r[i]$  залежить від відносних висот деяких  $k$  послідовних рослин.

Наприклад, нехай  $n = 5$ ,  $k = 3$  та  $i = 3$ . Наступні  $k - 1 = 2$  рослин за годинниковою стрілкою від рослини  $i = 3$  будуть рослина 4 та рослина 0. Якщо рослина 4 вища за рослину 3 та рослина 0 нижча за рослину 3, Хазель запише  $r[3] = 1$ .

Можна бути певними, що Хазель вірно записав значення  $r[i]$ . Отже, існує принаймні одна конфігурація різних висот рослин, що відповідає цим значенням.

Вам потрібно порівняти висоти  $q$  пар рослин. Нажаль, ви не можете потрапити на виставку. Ваше єдине джерело інформації це записник Хазела зі значенням  $k$  та послідовністю значень  $r[0], \dots, r[n - 1]$ .

Для кожної пари різних рослин  $x$  та  $y$ , що треба порівняти, визначіть, яка з наступних трьох ситуацій справджується:

- Рослина  $x$  точно вища за рослину  $y$ : у довільній конфігурації різних висот  $h[0], \dots, h[n - 1]$ , що відповідають масиву  $r$  маємо  $h[x] > h[y]$ .
- Рослина  $x$  точно нижча за рослину  $y$ : у довільній конфігурації різних висот  $h[0], \dots, h[n - 1]$ , що відповідають масиву  $r$  маємо  $h[x] < h[y]$ .
- Порівняння неоднозначне: жоден з попередніх двох випадків не виконується.

## Деталі реалізації

Ви маєте реалізувати наступні процедури:

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

- $k$ : кількість послідовних рослин, чиї висоти визначають кожне із значень  $r[i]$ .
- $r$ : масив розміру  $n$ , де  $r[i]$  це кількість рослин вищих за рослину  $i$  серед наступних  $k - 1$  рослин за годинниковою стрілкою.
- Ця процедура викликається рівно один раз перед викликами процедури

`compare_plants.`

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

- $x, y$ : номери рослин для порівняння.
- Ця процедура має повернути:
  - 1 якщо рослина  $x$  точно вища за рослину  $y$ ,
  - $-1$  якщо рослина  $x$  точно нижча за рослину  $y$ ,
  - 0 якщо порівняння неоднозначне.
- Ця процедура викликається  $q$  разів.

## Приклади

### Приклад 1

Розглянемо наступний виклик:

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

Нехай модуль перевірки викликає `compare_plants(0, 2)`. Оскільки  $r[0] = 0$  ми можемо одразу зробити висновок що рослина 2 не вища за рослину 0. Отже цей виклик має повернути 1.

Нехай після цього модуль перевірки викликає `compare_plants(1, 2)`. Для всіх можливих конфігурацій висот що задовольняють задані обмеження, рослина 1 нижча за рослину 2. Отже цей виклик має повернути  $-1$ .

### Приклад 2

Розглянемо наступний виклик:

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

Нехай модуль перевірки викликає `compare_plants(0, 3)`. Оскільки  $r[3] = 1$ , ми знаємо що рослина 0 вища за рослину 3. Отже цей виклик має повернути 1.

Нехай після цього модуль перевірки викликає `compare_plants(1, 3)`. Конфігурації висот  $[3, 1, 4, 2]$  та  $[3, 2, 4, 1]$  обидві відповідають записам Хазела. Оскільки рослина 1 нижча рослини 3 в одній конфігурації та вища ніж рослина 3 у іншій, цей виклик має повернути 0.

## Обмеження

- $2 \leq k \leq n \leq 200\,000$
- $1 \leq q \leq 200\,000$

- $0 \leq r[i] \leq k - 1$  (для всіх  $0 \leq i \leq n - 1$ )
- $0 \leq x < y \leq n - 1$
- Існують одна або більше конфігурацій **різних висот** рослин що відповідають масиву  $r$ .

## Підзадачі

1. (5 балів)  $k = 2$
2. (14 балів)  $n \leq 5000, 2 \cdot k > n$
3. (13 балів)  $2 \cdot k > n$
4. (17 балів) Вірна відповідь на кожен виклик `compare_plants` це 1 або  $-1$ .
5. (11 балів)  $n \leq 300, q \leq \frac{n \cdot (n-1)}{2}$
6. (15 балів)  $x = 0$  для кожного виклику `compare_plants`.
7. (25 балів) Без додаткових обмежень.

## Приклад модуля перевірки

Приклад модуля перевірки читає вхідні дані у наступному форматі:

- рядок 1:  $n \ k \ q$
- рядок 2:  $r[0] \ r[1] \ \dots \ r[n - 1]$
- рядок  $3 + i$  ( $0 \leq i \leq q - 1$ ):  $x \ y$  у  $i$ -му виклику `compare_plants`

Приклад модуля перевірки друкує ваші відповіді у наступному форматі:

- рядок  $1 + i$  ( $0 \leq i \leq q - 1$ ): значення, що повернув  $i$ -й виклик `compare_plants`.