

# Botanische Ausstellung (plants)

Die Botanikerin Hazel hat eine ungewöhnliche Ausstellung in den botanischen Gärten von Singapur besucht. Für diese Ausstellung wurden  $n$  Pflanzen **unterschiedlicher Höhen** in einem Kreis angeordnet. Sie sind im Uhrzeigersinn von 0 bis  $n - 1$  nummeriert, wobei Pflanze  $n - 1$  neben Pflanze 0 steht.

Während ihres Besuchs hat Hazel die  $i$ -te Pflanze ( $0 \leq i \leq n - 1$ ) mit den  $k - 1$  nachfolgenden verglichen, und sich die Anzahl  $r[i]$  notiert, wie viele dieser  $k - 1$  Pflanzen grösser als Pflanze  $i$  sind. Somit hängt  $r[i]$  von der relativen Höhe  $k$  aufeinanderfolgenden Pflanzen ab.

Beispielsweise sei  $n = 5$ ,  $k = 3$  und  $i = 3$ . Die  $k - 1 = 2$  nächsten Pflanzen im Uhrzeigersinn nach Pflanze  $i = 3$  sind somit Pflanze 4 und Pflanze 0. Wenn nun Pflanze 4 grösser ist als Pflanze 3 und Pflanze 0 kleiner, dann hätte sich Hazel  $r[3] = 1$  notiert.

Du darfst davon ausgehen, dass sich Hazel die Werte  $r[i]$  richtig notiert hat als sie in der Ausstellung war. Es gibt also mindestens eine Anordnung von Pflanzen welche mit diesen Werten von  $r[i]$  konsistent ist.

Du wurdest aufgefordert, die Höhen von  $q$  Pflanzenpaaren zu vergleichen. Selber warst du jedoch noch nicht in der Ausstellung und kennst somit nicht die Höhen der ausgestellten Pflanzen. Glücklicherweise hat dir Hazel ihr Notizbuch zur Verfügung gestellt, also sind dir die Werte von  $k$  und der Sequenz  $r[0], \dots, r[n - 1]$  bekannt.

Für jedes Paar verschiedener Pflanzen  $x$  und  $y$ , die du vergleichen sollst, bestimme, ob:

- Pflanze  $x$  sicher grösser ist als Pflanze  $y$ . Das heisst in jeder Anordnung von Pflanzen mit unterschiedlichen Höhen  $h[0], \dots, h[n - 1]$ , welche mit dem Array  $r$  konsistent ist, gilt  $h[x] > h[y]$
- Pflanze  $x$  sicher kleiner ist als Pflanze  $y$ . Das heisst in jeder Anordnung von Pflanzen mit unterschiedlichen Höhen  $h[0], \dots, h[n - 1]$ , welche mit dem Array  $r$  konsistent ist, gilt  $h[x] < h[y]$ .
- Der Vergleich nicht eindeutig entscheidbar ist, also keiner der vorigen Fälle eintritt.

## Implementierungsdetails

Du sollst folgenden Funktionen implementieren:

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

- $k$ : die Anzahl aufeinanderfolgender Pflanzen welche das Array  $r[i]$  bestimmen.

- $r$ : ein Array der Länge  $n$ , wobei  $r[i]$  die Anzahl ist, wie viele der  $k - 1$  im Uhrzeigersinn folgenden Pflanzen grösser als Pflanze  $i$  sind.
- Diese Funktion wird genau einmal aufgerufen, vor jedem Aufruf von `compare_plants`.

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

- $x, y$ : die Nummern der zu vergleichenden Pflanzen.
- Diese Funktion soll folgende Werte zurückgeben:
  - 1, wenn Pflanze  $x$  sicher grösser ist als Pflanze  $y$ ,
  - $-1$ , wenn Pflanze  $x$  sicher kleiner ist als Pflanze  $y$ ,
  - 0, wenn keiner der beiden vorherigen Fälle eintritt.
- Diese Funktion wird genau  $q$  mal aufgerufen.

## Beispiele

### Beispiel 1

Betrachte den folgenden Aufruf:

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

Angenommen, der Grader ruft `compare_plants(0, 2)` auf. Wegen  $r[0] = 0$  ist klar dass Pflanze 2 kleiner sein muss als Pflanze 0, da ansonsten  $r[0] > 0$  gelten müsste. Also sollte dieser Aufruf 1 zurückgeben.

Angenommen, der Grader ruft als nächstes `compare_plants(1, 2)` auf. Für jede mögliche Anordnung von Pflanzen, welche die obigen Bedingungen erfüllen, ist Pflanze 1 kleiner als Pflanze 2. Also sollte  $-1$  zurückgegeben werden.

### Beispiel 2

Betrachte den folgenden Aufruf:

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

Angenommen, der Grader ruft `compare_plants(0, 3)` auf. Wegen  $r[3] = 1$  wissen wir dass Pflanze 0 grösser ist als Pflanze 3. Also sollte dieser Aufruf 1 zurückgeben.

Angenommen, der Grader ruft als nächstes `compare_plants(1, 3)` auf. Die Anordnungen  $[3, 1, 4, 2]$  und  $[3, 2, 4, 1]$  sind beide konsistent mit Hazels Notizen. Weil in der ersten Anordnung Pflanze 1 kleiner ist als Pflanze 3, aber in der zweiten Anordnung grösser, ist der Vergleich nicht eindeutig entscheidbar. Es sollte also 0 zurückgegeben werden.

## Beschränkungen

- $2 \leq k \leq n \leq 200\,000$
- $1 \leq q \leq 200\,000$
- $0 \leq r[i] \leq k - 1$  (für alle  $0 \leq i \leq n - 1$ )
- $0 \leq x < y \leq n - 1$
- Es existiert mindestens eine Anordnung von Pflanzen **unterschiedlicher Höhen**, welche mit dem Array  $r$  konsistent ist.

## Teilaufgaben

1. (5 Punkte)  $k = 2$ .
2. (14 Punkte)  $n \leq 5000, 2 \cdot k > n$ .
3. (13 Punkte)  $2 \cdot k > n$ .
4. (17 Punkte) Der richtige Rückgabewert jedes Aufrufs von `compare_plants` ist entweder 1 oder  $-1$ .
5. (11 Punkte)  $n \leq 300, q \leq \frac{n \cdot (n-1)}{2}$ .
6. (15 Punkte)  $x = 0$  für jeden Aufruf von `compare_plants`.
7. (25 Punkte) Keine weiteren Beschränkungen.

## Beispiel-Grader

Der Beispiel-Grader liest die Eingabe im folgenden Format:

- Zeile 1:  $n \ k \ q$
- Zeile 2:  $r[0] \ r[1] \ \dots \ r[n - 1]$
- Zeile  $3 + i$  ( $0 \leq i \leq q - 1$ ):  $x \ y$  für den  $i$ -ten Aufruf von `compare_plants`

Der Beispiel-Grader gibt Deine Antworten im folgenden Format aus:

- Zeile  $1 + i$  ( $0 \leq i \leq q - 1$ ): Rückgabewert des  $i$ -ten Aufrufs von `compare_plants`.