

## Jamoaviy rasm (photo)

*Kecha 5 soatlik musobaqa davomida umuman suv ichmagan Qosimjonda gallyutsinatsiya boshlandi. Uning ko'zlariga har xil voqealar ko'rinmoqda.*

Qosimjonning maktabida  $k$  ta o'quvchi bor, ular 0 dan  $k - 1$  gacha raqamlangan. Gallyutsinatsiya natijasida Qosimjon hech qaysi o'quvchilarning bo'yi uzunligini eslay olmaydi, lekin u aniq biladi: hammaning bo'yi uzunligi har xil.

Yaqinda bitiruv kechasi bo'lib o'tadi va unda o'quvchilar jamoaviy rasmga tushishlari kerak.

Qosimjon uzunligi  $n$  bo'lgan  $a[0], a[1], \dots, a[n - 1]$  — o'quvchilarning rasmdagi tartibini (chapdan-o'ngga) yozib chiqdi. **Bunda ba'zi o'quvchilar rasmda bir-necha marta uchrashlari mumkin** – biroz g'alati bo'lsa ham, Qosimjon gallyutsinatsiya holatida ekanini unutmaymiz. Masala shartini yaxshiroq tushunish uchun namunalarga qarang.

Qosimjon sizga  $q$  marta quyidagi so'rovlarni bermoqchi:

- $a[l], a[l + 1], \dots, a[r]$  oraliqdagi o'quvchilarning bo'y uzunliklari alternatsiya qilishi mumkinmi? Ya'ni, agar  $k$  ta o'quvchilarning bo'ylari  $h[0], h[1], \dots, h[k - 1]$  bo'lsa,  $h[a[l]] > h[a[l + 1]] < h[a[l + 2]] > \dots$  bo'lishi mumkinmi?

E'tibor bering, bunda har bir so'rov uchun  $h[0], h[1], \dots, h[k - 1]$  sonlarni boshidan tanlash mumkin. Boshqacha qilib aytganda, barcha so'rovlar bir-biridan mustaqil.

Qosimjonning so'rovlariga bering.

## Implementation details

Siz quyidagi ikki protsedurani dasturlashingiz kerak:

```
void init(int k, int n, vector<int> a)
```

- $k$ : jami o'quvchilar soni.
- $n$ : rasmdagi jami odamlar soni.
- $a$ : uzunligi  $n$  bo'lgan massiv – odamlarning rasmdagi tartibi.
- Bu protsedura hech narsa qaytarmaydi.
- Bu protsedura aynan bir marta, barcha so'rovlardan oldin chaqiriladi.

```
bool query(int l, int r)
```

- $l, r$ : so'rovdagi oraliq.
- Bu protsedura `true` yoki `false` qaytarishi kerak.
- Bu protsedura aynan  $q$  marta chaqiriladi.

## Example

Quyidagi chaqiruvni ko'raylik:

```
init(3, 6, [0, 0, 1, 2, 0, 1])
```

Demak,  $k = 3, n = 6, a = [0, 0, 1, 2, 0, 1]$ .

Birinchi so'rov quyidagicha bo'lsin:

```
query(0, 1)
```

So'ralayotgan oraliq  $a[0 : 1] = [0, 0]$ .  $h$  ning hech qaysi qiymatlarida  $h[0] > h[0]$  shart bajarilmaydi. Protседura `false` qaytarishi kerak.

Ikkinchi so'rov:

```
query(1, 4)
```

So'ralayotgan oraliq  $a[1 : 4] = [0, 1, 2, 0]$ . Agar  $h[0] = 170\text{cm}, h[1] = 150\text{cm}, h[2] = 180\text{cm}$  bo'lsa,  $h[0] > h[1] < h[2] > h[0]$  shart bajariladi. Demak protsedura `true` qaytarishi kerak.

Uchinchi so'rov:

```
query(1, 5)
```

So'ralayotgan oraliq  $a[1 : 5] = [0, 1, 2, 0, 1]$ . Ko'rish mumkinki,  $h[0] > h[1] < h[2] > h[0] < h[1]$  shart bajarilishi uchun  $h[0] > h[1]$  va  $h[0] < h[1]$  shartlar bajarilishi kerak, bu esa ilojiy. Protседura `false` qaytarishi kerak.

## Constraints

- $2 \leq n \leq 4000$
- $2 \leq k \leq n$
- $1 \leq q \leq 10^6$
- $0 \leq a[i] \leq k - 1$  (barcha  $0 \leq i < n$  uchun)

- $0 \leq l < r \leq n - 1$  (barcha so'rovlar uchun)
- Barcha o'quvchilar rasmda istalgancha uchrashi mumkin (xususan, 0 marta ham).

## Subtasks

1. (12 ball)  $n, q \leq 300, k \leq \min(n, 5)$
2. (12 ball)  $k = 2$
3. (27 ball)  $n \leq 300$
4. (12 ball)  $q \leq 1000$
5. (26 ball)  $n \leq 1000$
6. (11 ball) Qo'shimcha cheklavlarsiz.

## Sample Grader

Namunaviy grader ma'lumotlarni quyidagi tartibda o'qiydi:

- qator 1:  $k \ n$
- qator 2:  $a[0] \ a[1] \ \dots \ a[n - 1]$
- qator 3:  $q$
- qator 4 +  $i$  ( $0 \leq i < q$ ):  $l \ r$

Har bir bajarilgan so'rov uchun, namunaviy grader javoblarni quyidagi tartibda chiqaradi:

- qator 1 +  $i$  ( $0 \leq i < q$ ): `query()` funksiyasi qaytargan javob: `true` bo'lsa 1, `false` bo'lsa 0.