

Problem Brperm

C header: brperm.h
C++ header: brperm.h

Notă: În enunț, $\overline{b_1 \dots b_k}$ reprezintă un întreg scris în notatie binară, unde b_1 este cel mai semnificativ bit, iar b_k este cel mai puțin semnificativ bit.

Vrăjitoarea Roxana, în timp ce zbura pe mătura prin galaxie, a descoperit o nouă planetă (planeta BR-PERM) unde toți locuitorii erau implicați într-un nou dans. În acest dans, participanții stau într-o linie, iar apoi se reordonează. Într-un dans la care participă 2^k locuitori, persoana de la poziția $\overline{b_1 \dots b_k}$ se va muta la poziția $\overline{b_k \dots b_1}$ (indexat de la 0).

Roxana a realizat că fiecare persoana de pe BR-PERM poartă îmbrăcăminte din una dintre cele 26 de culori. Aceste culori vor fi reprezentate de litere ale alfabetului Latin.

BR-PERM-ienii consideră speciale șirurile de dansatori unde secvența de culori pe care locuitorii le poartă înainte și după dans sunt la fel. Ei numesc astfel de secvențe *drăguțe*. De exemplu, când $k = 2$, avem un șir de 4 dansatori 0, 1, 2, 3, care după dans va fi ordonat în următorul fel: 0, 2, 1, 3. Astfel, secvența de culori *abba* este *drăguță*, dar *abca* nu este.

BR-PERM-ienii o roagă pe Roxana să îi ajute cu această problemă (se pare că vrăjitoarele mereu ajută oamenii să își rezolve problemele). Aceștia îi arată un șir de n dansatori și o roagă să îi raspunda la mai multe întrebări: "Este secvența de lungime 2^k care începe la dansatorul i *drăguță*?"

Protocol de interacțiune

Concurentul trebuie să implementeze două funcții, prima dintre ele fiind:

```
void init(int n, const char s[]);
```

Această funcție va fi apelată o singură dată, la începutul interacțiunii, și va furniza concurentului sirul de culori al hainelor dansatorilor, prin parametrul s .

Cea de-a doua funcție care trebuie implementată este:

```
int query(int i, int k);
```

Această funcție va fi apelată de exact Q ori și va returna 1 dacă și numai dacă subsecvența continuă de lungime 2^k din s care începe la al i -lea dansator (indexat de la 0), este *drăguță*. În caz contrar, va returna 0. Se garantează că subsecvența nu va depăși capătul dreapta al lui s .

Restricții

- $1 \leq N \leq 500\,000$
- $1 \leq Q \leq 500\,000$

Subtask 1 (13 puncte)

- $1 \leq N \leq 1\,000$
- $1 \leq Q \leq 1\,000$

Subtask 2 (37 de puncte)

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$

Subtask (17 puncte)

- s conține doar caracterele 'a' și 'b'.
- Culorile sunt alese aleator independent cu o anumită probabilitate fixata pentru fiecare test.

Subtask 4 (33 points)

- Fără restricții suplimentare.

Exemplu

input	output
<code>init(8, "axxyxyb")</code>	<code>query(0, 3) = true</code> <code>query(1, 1) = true</code> <code>query(0, 2) = false</code> <code>query(3, 2) = true</code>