

B. Dark Ride

Problem Name	Dark Ride
Time Limit	1 seconds
Memory Limit	1 gigabyte

Erika a obținut recent un loc de muncă pe timpul verii la parcul de distracții Phantasialand de lângă Bonn. Ea a fost angajată să controleze luminile din camerele prin care va trece o cursă în întuneric.

Cursa trece prin N camere, numerotate de la 0 la $N - 1$. Camerele sunt traversate în ordine, începând cu camera 0 și terminând cu camera $N - 1$. Luminile din camere sunt controlate de N întrerupătoare (de asemenea, numerotate de la 0 la $N - 1$), unul pentru fiecare cameră. Întrerupătorul s (unde $0 \leq s < N$) controlează lumina din camera p_s .

Șeful Erikăi a rugat-o să aprindă luminile în prima și ultima cameră și să le stingă pe toate celelalte. Sună ușor, nu-i așa? Ea trebuie doar să apese cele două întrerupătoare A și B pentru care $p_A = 0$ and $p_B = N - 1$ (or $p_B = 0$ and $p_A = N - 1$). Din păcate, Erika nu a fost pe deplin atentă când șeful ei i-a descris întrerupătoarele, și **nu își amintește șirul p - care descrie ce cameră controlează fiecare întrerupător.**

Erika trebuie să rezolve această situație înainte să observe șeful ei. Înainte de începutul fiecărei curse, Erika stinge toate luminile și poate apoi să apese o submulțime din întrerupătoare. Pe măsură ce cursa avansează dintr-o cameră în alta, oricând se trece dintr-o cameră iluminată în una neiluminată sau viceversa, Erika va auzii pasagerii țipând de entuziasm. Viteza călătoriei poate varia, așa că Erika nu poate să deducă direct ce camere sunt iluminate, însă va putea auzi numărul de țipete. Astfel, ea va afla de câte ori călătoria trece dintr-o cameră iluminată în una neiluminată, respectiv dintr-o cameră neiluminată în una iluminată.

O poți ajuta pe Erika să își dea seama care sunt întrerupătoarele care controlează luminile din prima și ultima cameră înainte ca șeful ei să observe? Poți să folosești cel mult 30 de călătorii.

Interaction

Aceasta este o problemă interactivă.

- Programul tău trebuie să înceapă prin a citi un număr întreg N : numărul de camere prin care trece cursa.

- Apoi, programul tău va trebui să interacționeze cu un grader. Pentru a începe o cursă, trebuie să afișezi o linie care conține un semn de întrebare "?", și apoi un șir de N caractere format din 0 (stins) și 1 (aprins), indicând configurația celor N întrerupătoare. Apoi, programul tău trebuie să citească un singur număr întreg ℓ ($0 \leq \ell < N$), reprezentând de câte ori Erika îi aude pe pasageri țipând.
- Când dorești să dai răspunsul, afișează o linie cu un semn al exclamării "!", urmat de două numere întregi A și B ($0 \leq A, B < N$). Pentru ca răspunsul tău să fie acceptat, aceste numere trebuie să fie indicii întrerupătoarelor care controlează cele două camere din capete, în orice ordine. După aceasta, programul ar trebui să se încheie.

Graderul nu este adaptiv, ceea ce înseamnă că șirul ascuns p este determinat înainte ca acțiunea să înceapă.

Nu uita să faci flush ieșirea standard după fiecare utilizare, altfel programul ar putea primi verdictul Time Limit Exceeded. În Python, aceasta se face automat dacă folosești `input()` să citești linii. În C++, `cout << endl;` face flush pe lângă afișarea unei linii noi; dacă utilizezi `printf`, trebuie să folosești `fflush(stdout)`.

Constraints and Scoring

- $3 \leq N \leq 30\,000$.
- Poți să folosești maxim 30 de curse (afișarea răspunsului final nu se consideră o cursă). Dacă depășești această limită, vei primi verdictul "Wrong Answer".

Soluția dumneavoastră va fi testată pe un set de grupuri de teste, fiecare valorând un număr de puncte. Fiecare grup de teste conține un set de cazuri de testare. Pentru a obține punctele pentru un grup de teste, trebuie să rezolvați toate testele din acea grupă.

Group	Score	Limits
1	9	$N = 3$
2	15	$N \leq 30$
3	17	$p_0 = 0$, i.e., întrerupătorul 0 controlează camera 0
4	16	N este par, cu întrerupătorul pentru una dintre camerele din capete situat în prima jumătate ($0 \leq A < \frac{N}{2}$) și celălalt întrerupător, în a doua jumătate ($\frac{N}{2} \leq B < N$)
5	14	$N \leq 1000$
6	29	Fără restricții adiționale

Testing Tool

Pentru a facilita testarea propriei soluții, am pus la dispoziție un tool simplu pe care îl poți descărca. Vezi "attachments" în partea de jos a paginii problemei din Kattis. Folosirea acestui tool este opțională. Reține că evaluatorul oficial Kattis diferă de cel de testare furnizat.

Pentru a utiliza toolul, creează un fișier de intrare, de exemplu „sample1.in”, care ar trebui să înceapă cu un număr N urmat de o linie cu p_0, p_1, \dots, p_{N-1} reprezentând permutarea ascunsă. De exemplu:

```
5
2 1 0 3 4
```

Pentru programe Python, de exemplu `solution.py` (în mod normal rulat cu `pypy3 solution.py`), execută:

```
python3 testing_tool.py pypy3 solution.py < sample1.in
```

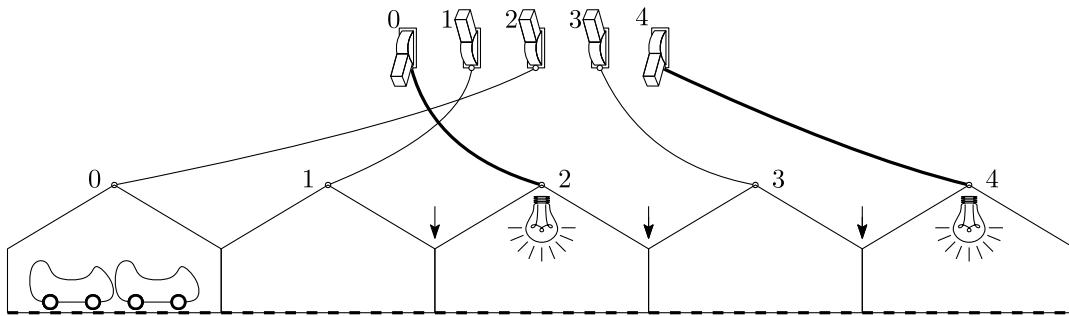
Pentru programe C++, mai întâi compilează (de exemplu cu `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`) și apoi rulează:

```
python3 testing_tool.py ./solution.out < sample1.in
```

Example

În primul exemplu, permutarea ascunsă este $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$. Aceasta îndeplinește constrângerile grupurilor de teste 2, 5 și 6.

Mai întâi, programul citește numărul întreg $N = 5$. Apoi, acesta solicită o cursă cu două întrerupătoare apăstate: întrerupătorul 4 și întrerupătorul 0. Acestea controlează camerele $p_4 = 4$ și $p_0 = 2$; vezi ilustrația de mai jos. Erika aude 3 țipete (marcate cu săgeți în figură): prima dată când cursa trece din camera neiluminată 1 în camera iluminată 2; a doua oară din camera iluminată 2 în camera neiluminată 3; și a treia oară când trece din camera neiluminată 3 în camera iluminată 4. Programul solicită apoi o altă cursă în care camerele p_0, p_2 și p_3 sunt iluminate, făcând-o pe Erika să audă 3 țipete. În cele din urmă, programul răspunde cu $A = 2$ și $B = 4$, ceea ce este într-adevăr corect, deoarece acestea controlează prima și ultima cameră ($p_2 = 0$ și $p_4 = 4$). Reține că $A = 4$ și $B = 2$ ar fi fost, de asemenea, o soluție corectă.



În al doilea exemplu, permutarea ascunsă este $[p_0, p_1, p_2] = [2, 0, 1]$. Aceasta satisface constrângerile grupurilor de teste 1, 2, 5, și 6. Programul solicită o cursă în care toate cele trei întrerupătoare sunt apăstate. Cum aceasta înseamnă că toate camerele sunt iluminate, Erika nu va auzi țipete. În a doua cursă, întrerupătoarele 1 și 0 sunt apăstate, ceea ce face ca luminile din camerele $p_1 = 0$ și $p_0 = 2$ să fie aprinse, în timp ce lumina din camera 1 este stinsă. Erika aude două țipete: când cursa merge din camera 0 (aprinsă) în camera 1 (stinsă) și din camera 1 (stinsă) în camera 2 (aprinsă). În ultima cursă, niciun întrerupător nu este apăsat, ceea ce înseamnă că luminile din toate cele trei camere sunt stinse și, din nou, Erika nu aude țipete. Programul răspunde apoi cu întrerupătoarele 1 și 0, care controlează într-adevăr prima și ultima cameră. Atât "1 0 1", cât și "1 1 0", reprezintă soluții acceptate.

În al treilea exemplu, permutarea ascunsă este $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$. Aceasta satisface constrângerile grupurilor de teste 2, 3, 4, 5, și 6. Reține că nu este neapărat posibil să deducem răspunsul după această singură cursă, dar soluția exemplu a ghicit răspunsul și a avut noroc.

First Sample

grader output	your output
5	
	? 10001
3	
	? 10110
3	
	! 2 4

Second Sample

grader output	your output
3	
	? 111
0	
	? 110
2	
	? 000
0	
	! 1 0

Third Sample

grader output	your output
4	
	? 1010
3	
	! 0 3