

В. Темно Возење

Име на проблемот	Темно Возење
Временско ограничување	1 секунда
Мемориско ограничување	1 гигабајт

Антонела неодамна доби летна работа во забавниот парк Фантазијаленд во близина на Бон. Таа беше ангажирана да ги контролира светлата во собите низ кои минува темно возење.

Возењето поминува низ N соби, нумерирани од 0 до $N - 1$. Собите се изминуваат по ред, почнувајќи од собата 0 и завршувајќи во собата $N - 1$. Светлата во собите се контролираат со N прекинувачи (исто така нумерирани од 0 до $N - 1$), по еден за секоја соба. Прекинувачот s (каде $0 \leq s < N$) го контролира светлото во собата p_s .

Шефот на Антонела ја замолил да ги вклучи светлата во првата и последната соба и да ги исклучи сите други. Звучи лесно, нели? Таа само треба да ги вклучи двата прекинувачи A и B така што $p_A = 0$ и $p_B = N - 1$ (или $p_B = 0$ и $p_A = N - 1$). За жал, Антонела не обрнала целосно внимание кога нејзиниот шеф ги опишал контролите и **таа не се сеќава на низата p – односно, кој прекинувач ја контролира која соба.**

Антонела треба да го реши ова пред да забележи нејзиниот шеф. Пред почетокот на секое возење, Антонела ги исклучува сите светла и потоа може да вклучи дел (подмножество) од прекинувачите. Како што возењето преминува од соба во соба, секогаш кога возењето ќе премине од осветлена соба во неосветлена соба или обратно, Антонела ќе ги чуе патниците како врескаат од возбуда. Брзината на возењето може да варира, па Антонела не може директно да заклучи кои соби се осветлени, но таа барем ќе го чуе бројот на вресоци. Со други зборови, таа ќе дознае колку пати возењето преминува од осветлена во неосветлена соба, или од неосветлена во осветлена соба.

Можете ли да ѝ помогнете на Антонела да открие кои два прекинувачи ги контролираат светлата за првата и последната соба пред да забележи нејзиниот шеф? Можете да искористите најмногу 30 возења.

Интеракција

Ова е интерактивен проблем.

- Вашата програма треба да започне со читање на линија со еден цел број N : бројот на соби во темното возење.
- Потоа, вашата програма треба да комуницира со оценувачот. За да започнете возење, треба да испечатите линија што започнува со прашалник „ ? “, а потоа и стринг со должина N што се состои од нули (0 - исклучено) и единици (1 - вклучено), што означува како сте ги поставиле N -те прекинувачи. Потоа, вашата програма треба да прочита еден цел број ℓ ($0 \leq \ell < N$), колку пати Антонела ги слуша патниците како врескаат.
- Кога сакате да одговорите, испечатете линија со извичник „ ! “, проследена со два цели броја A и B ($0 \leq A, B < N$). За да биде прифатен вашиот одговор, ова мора да бидат индексите на прекинувачите што ги контролираат двете крајни соби, по кој било редослед. После ова, вашата програма треба да заврши.

Оценувачот е неадаптивен, што значи дека скриената низа p се одредува пред да започне интеракцијата.

Осигурајте се дека го чистите (анг. flush) стандардниот излез по извршувањето на секое возење, во спротивно вашата програма може да биде оценета со Time Limit Exceeded. Во Python, ова се случува автоматски сè додека користите `input()` за читање линии. Во C++, `cout << endl;` активира чистење покрај тоа што печати нов ред; ако пак користите `printf`, искористете `fflush(stdout)`.

Ограничувања и бодување

- $3 \leq N \leq 30\,000$.
- Можете да иницирате најмногу до 30 возења (печатењето на конечниот одговор не се смета за возење). Доколку го надминете ова ограничување, ќе добиете оцена „Wrong Answer“.

Вашето решение ќе биде тестирано на множество од тест групи, пришто секоја од нив носи одреден број поени. Секоја тест група содржи множество од тест случаи. За да ги добиете поените за дадена тест група, треба да ги решите сите тест случаи во таа тест група.

Група	Поени	Ограничувања
1	9	$N = 3$
2	15	$N \leq 30$
3	17	$p_0 = 0$, т.е. прекинувачот 0 ја контролира собата 0
4	16	N е парен, пришто прекинувачот за една од крајните соби е во првата половина ($0 \leq A < \frac{N}{2}$), а оној за другата крајна соба е во втората половина ($\frac{N}{2} \leq B < N$)
5	14	$N \leq 1000$
6	29	Без дополнителни ограничувања

Алатка за тестирање

За да го олесниме тестирањето на вашето решение, ви обезбедивме едноставна алатка што можете да ја преземете. Видете ги „прилозите“ (анг. attachments) на дното од страницата со проблемите на системот Kattis. Алатката е опционална за употреба. Забележете дека официјалниот оценувач на Kattis се разликува од дадената алатка за тестирање.

За да ја искористите алатката, креирајте влезна датотека, како на пример „sample1.in“, која треба да започнува со број N , проследен со линија со p_0, p_1, \dots, p_{N-1} што ја специфицираат скриената пермутација. На пример:

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

За Python програми, да речеме `solution.py` (што вообичаено се извршува со `python3 solution.py`), извршете:

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

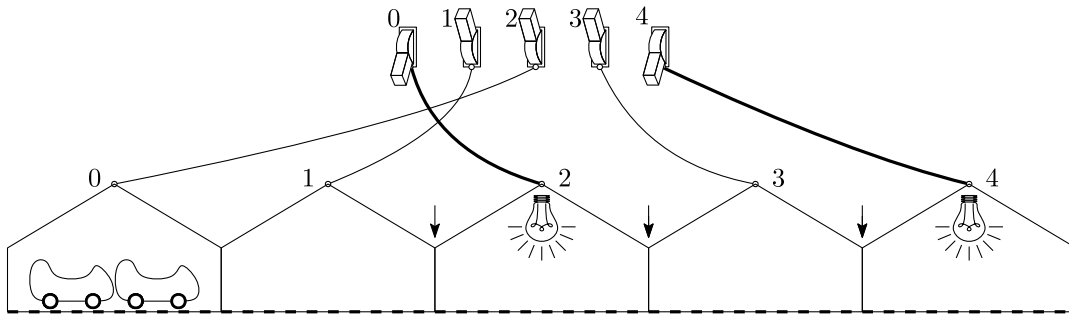
За C++ програми, прво искомпјилирајте (на пример, со `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`), а потоа извршете:

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

Пример

Во првиот пример, скриената пермутација е $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$. Ова ги задоволува ограничувањата на тест групите 2, 5 и 6. Прво, програмата го чита целиот број

$N = 5$. Потоа, програмата побарува возење со два вклучени прекинувачи: прекинувачот 4 и прекинувачот 0. Овие ги контролираат собите $p_4 = 4$ и $p_0 = 2$; погледнете ја илустрацијата подолу. Антонела слуша 3 вресоци (означени со стрелки на сликата): прво кога возењето преминува од неосветлената соба 1 во осветлената соба 2; второ, од осветлената соба 2 во неосветлената соба 3; и трето, кога преминува од неосветлената соба 3 во осветлената соба 4. Потоа програмата побарува друго возење кадешто собите p_0, p_2 и p_3 се осветлени, што ја принудува Антонела да слушне 3 вресоци. Конечно, програмата одговара со $A = 2$ и $B = 4$, што е навистина точно бидејќи овие ги контролираат првата и последната соба ($p_2 = 0$ и $p_4 = 4$). Забележете дека $A = 4$ и $B = 2$ исто така би биле точен одговор.



Во вториот пример, скриената пермутација е $[p_0, p_1, p_2] = [2, 0, 1]$. Ова ги задоволува ограничувањата на тест групите 1, 2, 5 и 6. Програмата побарува возење кадешто сите три прекинувачи се вклучени. Бидејќи ова значи дека сите соби се осветлени, Антонела нема да чуе вресоци. Во второто возење, прекинувачите 1 и 0 се вклучени, со што собите $p_1 = 0$ и $p_0 = 2$ се осветлени, додека собата 1 е неосветлена. Антонела слуша два вресоци: кога возењето преминува од собата 0 (осветлена) во собата 1 (неосветлена), и од собата 1 (неосветлена) во собата 2 (осветлена). Во последното возење, ниеден прекинувач не е вклучен, што значи дека сите соби се неосветлени, и дека повторно Антонела не слуша вресоци. Програмата потоа одговара со прекинувачите 1 и 0, кои навистина ги контролираат првата и последната соба. И „! 0 1“ и „! 1 0“ се одговори што се прифаќаат.

Во третиот пример, скриената пермутација е $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$. Ова ги задоволува ограничувањата на тест групите 2, 3, 4, 5 и 6. Да забележиме дека не мора да е возможно да се заклучи одговорот после ова единечно возење, но пример-решението имало среќа и го погодило одговорот.

Прв пример

излез од оценувач	ваш излез
5	
	? 10001
3	
	? 10110
3	
	! 2 4

Втор пример

излез од оценувач	ваш излез
3	
	? 111
0	
	? 110
2	
	? 000
0	
	! 1 0

Трет пример

излез од оценувач	ваш излез
4	
	? 1010
3	
	! 0 3