

## B. Sötét utazás

| Feladat neve  | Dark ride   |
|---------------|-------------|
| Időkorlát     | 1 másodperc |
| Memóriakorlát | 1 gigabyte  |

Erika nemrég nyári munkát kapott a Bonn melletti Phantasialand vidámparkban. Őt bízták meg azon szobák világításának irányításával, amelyeken a *Sötét utazás* attrakció áthalad.

Az utazás  $N$  szobán halad keresztül, amik  $0$ -tól  $N - 1$ -ig vannak számozva. A szobákon a számozás sorrendjében halad át, a  $0$ . szobától kezdve az  $N - 1$  sorszámú szobáig. A szobákban lévő lámpákat  $N$  kapcsoló vezérli (amik ugyancsak  $0$ -tól  $N - 1$ -ig vannak számozva), szobánként egy. Az  $s$  kapcsoló (ahol  $0 \leq s < N$ ) a  $p_s$  szoba világítását vezérli.

Erikát a főnöke megkérte, hogy kapcsolja fel a villanyt az első és az utolsó szobában és oltsa le az összes többiben. Egyszerűen hangzik, ugye? Csak fel kell kapcsolnia azon  $A$  és  $B$  kapcsolót, ahol  $p_A = 0$  és  $p_B = N - 1$  (vagy  $p_B = 0$  és  $p_A = N - 1$ ). Sajnos Erika nem figyelt oda teljesen, amikor a főnöke leírta a kapcsolók rendjét, és **nem emlékszik a  $p$  tömbre – azaz arra, hogy melyik kapcsoló melyik szobát vezérli.**

Erikának ki kell találnia ezt, mielőtt a főnöke észreveszi. Minden menet előtt Erika lekapcsolja az összes lámpát. Ezután bekapcsolja a kapcsolók egy részhalmazát. Ahogy a jármű szobáról szobára halad, valahányszor egy megvilágított szobából egy sötétbe ér, vagy fordítva, sötétből megvilágítottba ér, Erika hallani fogja az utasok izgatott sikolyát. A menet sebessége változhat, így Erika nem tudja közvetlenül kikövetkeztetni, hogy mely szobák vannak megvilágítva, de legalább a sikolyok számát hallja. Vagyis megtudja, hogy hányszor halad át a jármű egy megvilágított szobából egy sötétbe, vagy egy sötétből egy megvilágítottba.

Tudsz segíteni Erikának kitalálni, hogy melyik két kapcsoló vezérli az első és az utolsó szoba lámpáit, mielőtt a főnöke észrevenné, hogy nem tudja? Legfeljebb 30 menet információit használhatod fel.

## Interakció

Ez egy interaktív probléma.

- A programodnak egy  $N$  egész számot tartalmazó sor beolvasásával kell kezdődnie, ami a sötét menet szobáinak száma.
- Ezután a programodnak interakcióba kell lépnie az értékelővel. Egy menet elindításához egy sorba kérdőjellel ( "?") kezdve egy  $N$  hosszúságú, 0 -kból (kikapcsolva) és 1 -ekből (bekapcsolva) álló karakterláncot kell írnod, ami az  $N$  kapcsoló beállítását mutatja. Ezután a programodnak egyetlen  $\ell$  egész számot kell beolvasnia, ( $0 \leq \ell < N$ ), ami azt jelenti, hogy Erika hányszor hallotta az utasok sikolyát.
- Amikor válaszolni szeretnél, írd ki egy sort egy felkiáltójellel ( "!" ) kezdve, majd két egész  $A$  és  $B$  ( $0 \leq A, B < N$ ) számot, ami a két szélső szobát vezérlő kapcsolók indexei, tetszőleges sorrendben. Ezt követően a programnak be kell fejeződnie.

Az értékelő nem adaptív, ami azt jelenti, hogy a  $p$  rejtett tömböt az interakció megkezdése előtt meghatározzák.

Minden sor kiírása után feltétlenül ürítsd ki a standard kimenetet, különben a programod időkorlát túllépésként lesz értékelve. Pythonban ez automatikusan megtörténik, amíg az input() függvényt használsz a sorok beolvasásához.

C++-ban a `cout << endl;` függvény az új sor kiírása mellett ürít is; printf függvény használata esetén használd az `fflush(stdout)` parancsot.

## Korlátok és pontozás

- $3 \leq N \leq 30\,000$ .

A megoldásodat tesztcsoportokra teszteljük, minden tesztcsoport adott pontot ér. Minden tesztcsoport több tesztesetet tartalmaz. Egy tesztcsoport pontjainak megszerzéséhez az adott tesztcsoport összes tesztesetére helyesen kell futnia a megoldásodnak.

| Csoport | Pontszám | Korlátok   |
|---------|----------|--|
| 1       | 9        | $N = 3$  |
| 2       | 15       | $N \leq 30$  |
| 3       | 17       | $p_0 = 0$ , azaz a 0. kapcsoló vezérli a 0. szobát   |
| 4       | 16       | $N$ páros, az egyik szélső szoba a menet első felében ( $0 \leq A < \frac{N}{2}$ ), a másiké pedig a második felében ( $\frac{N}{2} \leq B < N$ ) van. |
| 5       | 14       | $N \leq 1000$  |
| 6       | 29       | Nincsenek további korlátok   |

## Tesztelő keretrendszer

A megoldás tesztelésének megkönnyítésére egy egyszerű alkalmazást biztosítunk, amelyet letölthetsz a Kattis probléma oldal alján található „attachments” menüpontnál. Az eszköz használata nem kötelező. Vedd figyelembe, hogy a Kattis hivatalos értékelője eltér a tesztelő eszköztől.

Az eszköz használatához hozz létre egy bemeneti fájlt, például a „sample1.in” fájlt, amely tartalmazza azt az  $N$  számot, majd a következő sorban a  $p_0, p_1, \dots, p_{N-1}$  értékeket, amellyel a programod tesztelni fogod.

Például:

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

**Python** programok esetén add ki a `solution.py` (általában `pypy3 solution.py`) utasítást, majd futtasd:

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

**C++** használata esetén először fordítsd le a programod (pl. `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`), majd futtasd a tesztelő eszközt a következő paranccsal:

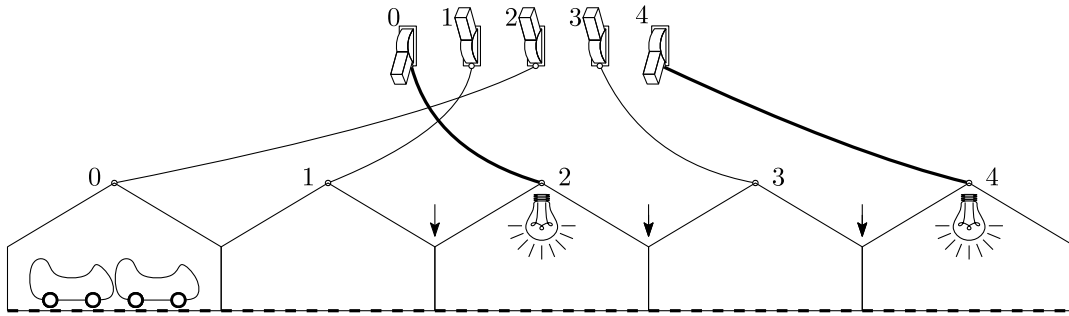
```
python3 testing_tool.py ./solution.out < sample1.in
```

## Példák

Az első példában a rejtett permutáció:  $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$ .

Ez kielégíti a 2., az 5. és a 6. tesztcsoport feltételeit.

A program beolvassa az  $N = 5$  értéket. Ezután egy menetet indít két felkapcsolt kapcsolóval, amelyek a 4. és a 0. kapcsolók. Ezek a kapcsolók a  $p_4 = 4$ -es szobát és a  $p_0 = 2$ -es szobát irányítják; lásd az ábrát. Erika 3 sikolyt hall (az ábrán nyíllal jelölve): először, amikor a menet a sötét 1-es szobából a kivilágított 2-es szobába ér; másodszor, amikor a kivilágított 2-es szobából a sötét 3-as szobába ér; és harmadikat, amikor a sötét 3-as szobából a kivilágított 4-esbe ér. A program ezután egy másik menetet indít, ahol a  $p_0, p_2$  és a  $p_3$  van felkapcsolva, és Erika 3 sikolyt hall. Végül a program az  $A = 2$  és a  $B = 4$  értékkel válaszol, ami valóban helyes, mivel ezek irányítják az első és az utolsó szobát ( $p_2 = 0$  és  $p_4 = 4$ ). Megjegyzés: az  $A = 4$  és  $B = 2$  is helyes válasz lett volna.



A második példában a rejtett permutáció:  $[p_0, p_1, p_2] = [2, 0, 1]$ . Ez kielégíti az 1., a 2., az 5. és a 6. tesztcsoport feltételeit. A program először mind a három kapcsolót felkapcsolja. Mivel ez azt jelenti, hogy mindhárom szoba kivilágított, így Erika nem hall sikolyt. A második menetben az 1. és a 0. kapcsoló van felkapcsolva, ezzel a  $p_1 = 0$  és a  $p_0 = 2$  szobák kivilágítottak, míg az 1. szoba sötét. Erika két sikolyt hall: amikor a menet a 0. kivilágított szobából az 1. sötét szobába ér, amikor a sötét 1. szobából a kivilágított 2. szobába ér. Az utolsó menetben egyetlen kapcsoló sincs felkapcsolva, ami azt jelenti, hogy mindhárom szoba sötét és így Erika nem hall sikolyt. A program az 1. és a 0. kapcsolót, ami az első és az utolsó szobát vezérli. Mind a "0 1" és a "1 0" válasz is megfelelő.

A harmadik példában a rejtett permutáció:  $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$ . Ez kielégíti a 2., a 3., a 4., az 5. és a 6. tesztcsoport feltételeit is. Megjegyzés: nem feltétlenül lehet kikövetkeztetni a választ ezen egyetlen út után, de a mintamegoldás kitalálta a választ és szerencséje volt.

## Első példa

| értékelő kimenete | saját kimenet |
|-------------------|---------------|
| 5                 |               |
|                   | ? 10001       |
| 3                 |               |
|                   | ? 10110       |
| 3                 |               |
|                   | ! 2 4         |

## Második példa

| értékelő kimenete | saját kimenet |
|-------------------|---------------|
| 3                 |               |
|                   | ? 111         |
| 0                 |               |
|                   | ? 110         |
| 2                 |               |
|                   | ? 000         |
| 0                 |               |
|                   | ! 1 0         |

## Harmadik példa

| értékelő kimenete | saját kimenet |
|-------------------|---------------|
| 4                 |               |
|                   | ? 1010        |
| 3                 |               |
|                   | ! 0 3         |