

## B. Dark Ride

Naam taak	Dark Ride
Tijdslimiet	1 seconde
Geheugenlimiet	1 gigabyte

Erika heeft onlangs een zomerbaan gekregen in het attractiepark Phantasialand bij Bonn. Ze wordt ingehuurd om de verlichting te bedienen in de ruimtes waar een darkride doorheen rijdt.

De rit gaat door  $N$  kamers, genummerd van  $0$  tot  $N - 1$ . De kamers worden in volgorde doorkruist, beginnend in kamer  $0$  en eindigend in kamer  $N - 1$ . De verlichting in de kamers wordt bediend door  $N$  schakelaars (eveneens genummerd van  $0$  tot  $N - 1$ ), één voor elke kamer. Schakelaar  $s$  (waarbij  $0 \leq s < N$ ) bedient de verlichting in kamer  $p_s$ .

Erika's baas heeft haar gevraagd de lichten in de eerste en laatste kamer aan te doen en alle andere uit te doen. Klinkt makkelijk, toch? Ze hoeft alleen de twee schakelaars  $A$  en  $B$  aan te zetten, zodat  $p_A = 0$  en  $p_B = N - 1$  (of  $p_B = 0$  en  $p_A = N - 1$ ). Helaas heeft Erika niet goed opgelet toen haar baas de bediening beschreef, en **herinnert ze zich de lijst  $p$  niet – dat wil zeggen, welke schakelaar welke kamer bedient**.

Erika moet dit uitzoeken voordat haar baas het merkt. Voor het begin van elke rit schakelt Erika alle lichten uit en vervolgens kan ze een deel van de schakelaars aanzetten. Terwijl de rit van kamer naar kamer gaat, hoort Erika de passagiers opgewonden schreeuwen wanneer de rit van een verlichte naar een onverlichte kamer gaat of andersom. De snelheid van de rit kan variëren, dus Erika kan niet direct afleiden welke kamers verlicht zijn, maar ze hoort in ieder geval het aantal schreeuwen. Dat wil zeggen, ze leert hoe vaak de rit van een verlichte naar een onverlichte kamer gaat, of van een onverlichte naar een verlichte kamer.

Kun jij Erika helpen uit te zoeken welke twee schakelaars de verlichting in de eerste en de laatste kamer bedienen voordat haar baas het merkt? Je mag maximaal 30 ritten gebruiken.

## Interactie

Dit is een interactief probleem.

- Je programma moet beginnen met het lezen van een regel met een geheel getal  $N$ : het aantal kamers in de darkride.

- Vervolgens moet je programma met de grader communiceren. Om een rit te starten, moet je een regel printen die begint met een vraagteken " ? " en vervolgens een string van lengte  $N$  bestaande uit 0'en (uitgeschakeld) en 1'en (ingeschakeld), die aangeven hoe je de  $N$  schakelaars instelt. Dan moet je programma één enkel geheel getal  $\ell$  ( $0 \leq \ell < N$ ) inlezen, het aantal keren dat Erika de passagiers hoort schreeuwen.
- Wanneer je wil antwoorden, print dan een regel met een uitroepteken " ! ", gevolgd door twee integers  $A$  en  $B$  ( $0 \leq A, B < N$ ). Om je antwoord te accepteren, moeten dit de indexen zijn van de schakelaars die de twee kamers aan de uiteindes bedienen, in een willekeurige volgorde. Hierna zou je programma moeten stoppen.

De grader is niet-adaptief, wat betekent dat de verborgen lijst  $p$  wordt bepaald voordat de interactie begint.

Zorg ervoor dat je de standaardoutput flusht na het aanvragen van elke rit, anders kan je programma worden beoordeeld als Time Limit Exceeded. In Python gebeurt dit automatisch zolang je `input()` gebruikt om regels te lezen. In C++ zorgt `cout << endl;` ervoor dat er wordt geflusht en dat er een nieuwe regel wordt afgedrukt; als je `printf` gebruikt, gebruik je `fflush(stdout)`.

## Randvoorwaarden en puntentelling

- $3 \leq N \leq 30\,000$ .
- Je kan maximaal 30 ritten aanvragen (het printen van het uiteindelijke antwoord telt niet als een rit). Als je boven dit limiet gaat, krijg je het oordeel "Wrong Answer".

Je oplossing wordt getest op een set van testgroepen, die elk een aantal punten waard zijn. Elke testgroep bevat een set van testgevallen. Om de punten voor een testgroep te krijgen, moet je alle testgevallen in de testgroep oplossen.

Group	Punten	Limieten
1	9	$N = 3$
2	15	$N \leq 30$
3	17	$p_0 = 0$ , oftewel, schakelaar 0 bedient kamer 0
4	16	$N$ is even, met de schakelaar voor één van de twee kamers aan de uiteindes in de eerste helft ( $0 \leq A < \frac{N}{2}$ ) en de andere in de tweede helft ( $\frac{N}{2} \leq B < N$ )
5	14	$N \leq 1000$
6	29	Geen aanvullende beperkingen

## Testing Tool

Om het testen van je oplossing makkelijker te maken, is er een eenvoudige tool beschikbaar die je kan downloaden. Zie "attachments" onderaan de Kattis-taakpagina. Het gebruik van deze tool is optioneel. Houd er rekening mee dat de officiële Kattis-grader anders is dan de meegeleverde testtool.

Om de tool te gebruiken, maak je een invoerbestand aan, zoals "sample1.in", dat begint met een getal  $N$  gevolgd door een regel met  $p_0, p_1, \dots, p_{N-1}$  die de verborgen permutatie specificeert. Bijvoorbeeld:

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

Voor Python-programma's, zoals `solution.py` (normaal gesproken uitgevoerd als `pypy3 solution.py`), voer het volgende uit:

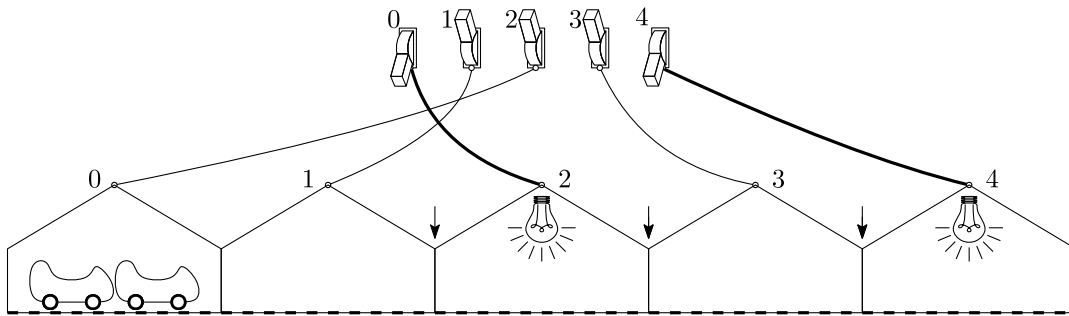
```
python3 testing_tool.py pypy3 solution.py < sample1.in
```

Voor C++-programma's moet je het eerst compileren (bijvoorbeeld met `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`) en voer dan uit:

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

## Voorbeeld

In het eerste voorbeeld is de verborgen permutatie  $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$ . Dit voldoet aan de beperkingen van testgroepen 2, 5 en 6. Eerst leest het programma het integer  $N = 5$ . Vervolgens vraagt het programma een rit aan met twee ingeschakelde schakelaars: schakelaar 4 en schakelaar 0. Deze bedienen kamers  $p_4 = 4$  en  $p_0 = 2$ ; zie de onderstaande illustratie. Erika hoort 3 keer geschreeuw (aangegeven met pijlen in de afbeelding): eerst wanneer de rit van onverlichte kamer 1 naar verlichte kamer 2 gaat; ten tweede van verlichte kamer 2 naar onverlichte kamer 3; en ten derde wanneer de rit van onverlichte kamer 3 naar verlichte kamer 4 gaat. Het programma vraagt dan om een andere rit waarbij de kamers  $p_0, p_2$  en  $p_3$  verlicht zijn, waardoor Erika 3 keer geschreeuw hoort. Ten slotte antwoordt het programma met  $A = 2$  en  $B = 4$ , wat inderdaad correct is aangezien deze de eerste en laatste kamer besturen ( $p_2 = 0$  en  $p_4 = 4$ ). Merk op dat  $A = 4$  en  $B = 2$  ook een correct antwoord zou zijn geweest.



In het tweede voorbeeld is de verborgen permutatie  $[p_0, p_1, p_2] = [2, 0, 1]$ . Dit voldoet aan de beperkingen van testgroepen 1, 2, 5 en 6. Het programma vraagt om een rit waarbij alle drie de schakelaars zijn ingeschakeld. Omdat dit betekent dat alle kamers verlicht zijn, hoort Erika geen geschreeuw. In de tweede rit worden schakelaars 1 en 0 ingeschakeld, waardoor kamers  $p_1 = 0$  en  $p_0 = 2$  verlicht zijn, terwijl kamer 1 onverlicht is. Erika hoort twee keer geschreeuw: wanneer de rit van kamer 0 (verlicht) naar kamer 1 (onverlicht) gaat, en van kamer 1 (onverlicht) naar kamer 2 (verlicht). In de laatste rit worden geen schakelaars ingeschakeld, wat betekent dat alle drie de kamers onverlicht zijn, en dat Erika opnieuw geen geschreeuw hoort. Het programma antwoordt vervolgens met schakelaars 1 en 0, die inderdaad de eerste en de laatste kamer bedienen. Zowel " ! 0 1 " als " ! 1 0 " zijn geaccepteerde antwoorden.

In het derde voorbeeld is de verborgen permutatie  $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$ . Dit voldoet aan de beperkingen van testgroepen 2, 3, 4, 5 en 6. Merk op dat het kan gebeuren dat het niet mogelijk is om het antwoord na deze ene rit te bepalen, maar de voorbeeldoplossing gokte het antwoord en had geluk.

### Eerste voorbeeld

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

## Tweede voorbeeld

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

## Derde voorbeeld

grader output	jouw output
4	
	? 1010
3	
	! 0 3