

## B. Dark Ride

Problemnamn	Dark Ride
Tidsgräns	1 sekunder
Minnesgräns	1 gigabyte

Erika fick nyligen ett sommarjobb på nöjesparken Phantasialand nära Bonn. Hon anställdes för att styra lamporna i rummen som åkattraktionen Dark Ride passerar genom.

Åkattraktionen går genom  $N$  rum, numrerade från 0 till  $N - 1$ . Rummen passeras i ordning från rum 0 till rum  $N - 1$ . Belysningen i rummen styrs av  $N$  strömbrytare (också numrerade från 0 till  $N - 1$ ), en för varje rum. Strömbrytare  $s$  (där  $0 \leq s < N$ ) styr ljuset i rum  $p_s$ .

Erikas chef har bett henne tända lamporna i det första och sista rummet och släcka alla andra. Låter enkelt, eller hur? Hon behöver bara slå på de två strömbrytarna  $A$  och  $B$  så att  $p_A = 0$  och  $p_B = N - 1$  (eller  $p_B = 0$  och  $p_A = N - 1$ ). Tyvärr var Erika inte helt uppmärksam när hennes chef beskrev kontrollerna och **hon kommer inte ihåg listan  $p$  - det vill säga vilken strömbrytare som styr vilket rum.**

Nu måste Erika reda ut det här innan hennes chef märker något. Inför varje åktur släcker Erika alla lampor. Därefter kan hon slå på en delmängd av strömbrytarna. Åkturen går sedan från rum till rum, och när den går från ett upplyst rum till ett mörkt rum eller vice versa kommer Erika höra passagerarna skrika av upphetsning. Åkturens hastighet kan variera, så Erika kan inte dra några slutsatser om vilka rum som är upplysta, men hon kommer åtminstone att höra antalet skrik. Det vill säga, hon kommer att få veta hur många gånger åkturen går från ett upplyst rum till ett mörkt rum, eller från ett mörkt till ett upplyst rum.

Kan du hjälpa Erika att lista ut vilka två strömbrytare som styr belysningen i det första och sista rummet innan hennes chef märker något? Du kan använda högst 30 åkturer.

## Interaktion

Detta är ett interaktivt problem.

- Ditt program ska börja med att läsa en rad med ett heltal  $N$ : antalet rum i Dark Ride.
- Sedan ska ditt program interagera med gradern. För att starta en åktur ska du skriva ut en rad som börjar med ett frågetecken " ? " och därefter innehåller en sträng av längd  $N$

bestående av 0 :or (avslagen) och 1 :or (påslagen), som anger hur du ställer in de  $N$  strömbrytarna. Efter det ska ditt program läsa ett heltal  $\ell$  ( $0 \leq \ell < N$ ), antalet gånger Erika hör passagerarna skrika.

- När du vill svara ska du skriva en rad med ett utropstecken " ! " följt av två heltal  $A$  och  $B$  ( $0 \leq A, B < N$ ). För att ditt svar ska accepteras måste dessa vara indexen för strömbrytarna som styr de två rummen i ändarna, i valfri ordning. Efter det ska ditt program avslutas.

Gradern är icke-adaptiv, vilket innebär att den okända listan  $p$  bestäms innan interaktionen börjar.

Se till att flusha standard output efter varje körning, annars kan ditt program bedömas som Time Limit Exceeded. I Python sker detta automatiskt så länge du använder `input()` för att läsa rader. I C++ skriver `cout << endl;` både ut en ny rad och flushar outputen; om du använder `printf`, använd `fflush(stdout)`.

## Begränsningar och poängsättning

- $3 \leq N \leq 30\,000$ .
- Du kan använda högst 30 åkturer (att skriva ut det slutliga svaret räknas inte som en åktur). Om du överskrider denna gräns får du "Wrong Answer".

Din lösning kommer att testas på flera testgrupper, som var och en är värd ett antal poäng. Varje testgrupp innehåller en mängd testfall. För att få poängen för en testgrupp måste du lösa alla testfall i testgruppen.

Grupp	Poäng	Begränsningar
1	9	$N = 3$
2	15	$N \leq 30$
3	17	$p_0 = 0$ , alltså att strömbrytare 0 styr rum 0
4	16	$N$ är jämnt, och för rummet i ena änden är brytaren i första halvan ( $0 \leq a < \frac{N}{2}$ ) och för det i andra änden är brytaren i andra halvan ( $\frac{N}{2} \leq b < N$ )
5	14	$N \leq 1000$
6	29	Inga ytterligare begränsningar

## Testverktyg

För att underlätta testningen av din lösning erbjuder vi ett enkelt verktyg som du kan ladda ner. Se "attachments" längst ner på Kattis problemsida. Verktyget är valfritt att använda. Observera att den officiella Kattis-gradern skiljer sig från det erbjudna testverktyget.

För att använda verktyget, kan du skapa en indatafil, till exempel "sample1.in", som ska börja med ett tal  $N$  följt av en rad med  $p_0, p_1, \dots, p_{N-1}$  som anger den dolda permutationen. Till exempel:

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

För Python-program, till exempel `solution.py` (som brukar köras med `python3 solution.py`), kör:

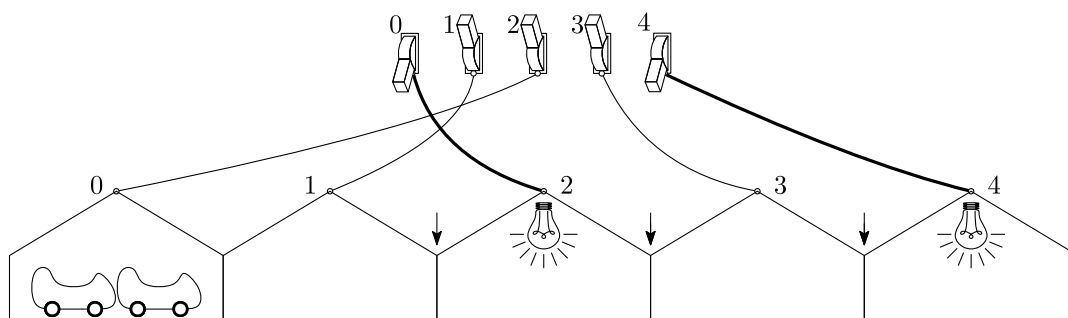
```
python3 testing_tool.py python3 solution.py < sample1.in
```

För C++-program, kompilera först (till exempel med `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`) och kör sedan:

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

## Exempelfall

I det första exemplet är den okända permutationen  $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$ . Detta uppfyller begränsningarna för testgrupperna 2, 5 och 6. Först läser programmet heltalet  $N = 5$ . Sedan begär programmet en åktur med  $K = 2$  påslagna brytare: brytare 4 och brytare 0. Dessa styr rum  $p_4 = 4$  och  $p_0 = 2$ ; se illustrationen nedan. Erika hör 3 skrik (markerade med pilar i figuren): först när åkattraktionen passerar från det mörka rummet 1 till det upplysta rummet 2; sedan från det upplysta rummet 2 till det mörka rummet 3; och tredje gången när den passerar från det mörka rummet 3 till det upplysta rummet 4. Programmet begär sedan en annan åktur där rummen  $p_0, p_2$  och  $p_3$  är upplysta, vilket gör att Erika hör 3 skrik. Slutligen svarar programmet med  $A = 2$  och  $B = 4$ , vilket alltså är korrekt eftersom de styr det första och sista rummet ( $p_2 = 0$  och  $p_4 = 4$ ). Observera att  $A = 4$  och  $B = 2$  också skulle ha varit ett korrekt svar.



I det andra exemplet är den dolda permutationen  $[p_0, p_1, p_2] = [2, 0, 1]$ . Detta uppfyller begränsningarna för testgrupperna 1, 2, 5 och 6. Programmet begär en åktur där alla tre strömbrytare är påslagna. Eftersom detta innebär att alla rum är upplysta, kommer Erika inte att höra några skrik. I den andra åkturen är strömbrytarna 1 och 0 påslagna, vilket gör att rummen

$p_1 = 0$  och  $p_0 = 2$  är upplysta, medan rum 1 är släckt. Erika hör två skrik: när åkturen går från rum 0 (upplyst) till rum 1 (släckt), och från rum 1 (släckt) till rum 2 (upplyst). I den sista åkturen är inga strömbrytare påslagna, vilket innebär att alla tre rummen är släckta, och återigen att Erika inte hör några skrik. Programmet svarar sedan med strömbrytarna 1 och 0, som faktiskt styr det första och sista rummet. Både " ! 0 1 " och " ! 1 0 " är giltiga svar att skriva ut.

I det tredje exemplet är den dolda permutationen  $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$ . Detta uppfyller begränsningarna för testgrupperna 2, 3, 4, 5 och 6.

### Första exempelfallet

graderns output	din output
5	
	? 10001
3	
	? 10110
3	
	! 2 4

### Andra exempelfallet

graderns output	din output
3	
	? 111
0	
	? 110
2	
	? 000
0	
	! 1 0

### Tredje exempelfallet

graderns output	din output
4	
	? 1010
3	
	! 0 3