

B. Jízda na černo (Dark ride)

Název úlohy	Jízda na černo (Dark Ride)
Časový limit	1 sekunda
Paměťový limit	1 gigabajt

Erika si nedávno našla letní práci v zábavním parku Phantasialand kousek od Bonnu. Dostala za úkol starat se o světla v místnostech, kterými projíždí černá jízda.

Jízda projíždí N místnostmi, očíslovanými od 0 do $N - 1$ včetně. Jízda začíná místností 0 a končí v místnosti $N - 1$. Světla v místnostech jsou ovládána N spínači (také očíslovanými od 0 do $N - 1$), jedním pro každou místnost. Spínač s (kde $0 \leq s < N$) ovládá světlo v místnosti p_s .

Eričin šéf ji požádal, aby rozsvítila světla v první a poslední místnosti a zhasla všechny ostatní. Zní to jednoduše, že? Stačí, když zapne dva vypínače A a B tak, aby $p_A = 0$ a $p_B = N - 1$ (nebo $p_B = 0$ a $p_A = N - 1$). Erika je bohužel správná Jarošanda, a nedávala pozor, když jí šéf popisoval ovládání, a **nepamatuje si pole p – tedy který vypínač ovládá kterou místnost**.

Erika to musí vyřešit dříve, než si toho všimne její šéf! Na začátku každé jízdy Erika zhasne všechna světla. Poté může zapnout jakoukoli podmnožinu spínačů. Při jízdě z místnosti do místnosti, kdykoli se jízda přesune z osvětlené do neosvětlené místnosti nebo naopak, uslyší Erika cestující křičet obdivem. Rychlost jízdy se může lišit, takže Erika nemůže přímo odvodit, které místnosti jsou osvětlené, ale alespoň uslyší počet křiků. To znamená, že se dozví, kolikrát jízda projede z osvětlené do neosvětlené místnosti nebo z neosvětlené do osvětlené místnosti.

Pomož Erice zjistit, které dva přepínače ovládají světla v první a poslední místnosti, než si toho všimne její šéf! Máš na to maximálně 30 jízd.

Interakce

Toto je interaktivní úloha.

- Váš program by měl nejdříve načíst řádek s celým číslem N : počtem místností v temné jízdě.
- Poté by měl váš program interagovat s hodnotičem. Pro spuštění jízdy byste měli vypsat řádek začínající otazníkem „ ? “ a poté řetězec délky N sestávající z 0 (vypnuto) a 1 (zapnuto), které označují, jak nastavíte přepínače N . Váš program by pak měl přečíst jedno celé číslo ℓ ($0 \leq \ell < N$), což je počet křiků, které Erika při jízdě uslyší.

- Pokud chcete odpovědět, vypíšte řádek s vykřičníkem „ ! “, za nímž následují dvě celá čísla A a B ($0 \leq A, B < N$). Aby byla vaše odpověď přijata, musí se jednat o indexy spínačů ovládajících dvě koncové místnosti, v libovolném pořadí. Poté by se měl váš program ukončit.

Grader není adaptivní, což znamená, že skryté pole p je určeno před zahájením interakce.

Po vydání každé jízdy nezapomeňte flushnout standardní výstup, jinak by váš program mohl být vyhodnocen jako `Time Limit Exceeded`. V Pythonu se to děje samo, pokud používáte `input()` pro čtení řádků. V C++ `cout << endl;` kromě výpisu znaku nového řádku také flushne; pokud používáte `printf`, použijte `fflush(stdout)`.

Omezení a bodování

- $3 \leq N \leq 30\,000$.
- Můžete zkusit maximálně 30 jízd (vypsání konečné odpovědi se nepočítá jako jízda). Pokud tento limit překročíte, dostanete verdikt „Wrong answer“.

Vaše řešení bude otestováno na několika sadách testů, z nichž každá má určitý počet bodů. Každá sada testů obsahuje několik testovacích případů. Abyste získali body za sadu testů, musíte vyřešit všechny testovací případy v dané sadě testů.

Sada	Body	Omezení
1	9	$N = 3$
2	15	$N \leq 30$
3	17	$p_0 = 0$, tedy přepínač 0 ovládá místnost 0
4	16	N je liché a přepínač jedné z krajních místností je v první polovině ($0 \leq a < \frac{N}{2}$) a druhý v druhé polovině ($\frac{N}{2} \leq b < N$)
5	14	$N \leq 1000$
6	29	Žádná další omezení

Testovátka

Pro usnadnění testování jsme pro vás připravili jednoduché testovátka, které si můžete stáhnout. Najdete je v sekci „attachments“ ve spodní části stránky úlohy v Kattis. Použití nástroje je dobrovolné. Upozorňujeme, že oficiální systém hodnocení Kattis se liší od poskytnutého testovacího nástroje.

Chcete-li nástroj použít, vytvořte vstupní soubor, například „hrošík.in“, který má na prvním řádku N , a na následujícím řádku p_0, p_1, \dots, p_{N-1} určujícím skrytou permutaci p . Například:

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

Pro programy v Pythonu, například `boucník.py` (obvykle spouštěné pomocí `python3 boucník.py`), spusťte:

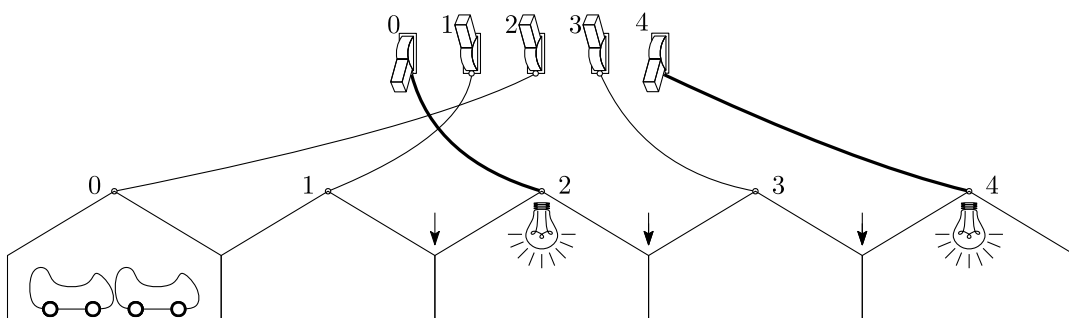
```
python3 testing_tool.py python3 boucník.py < hrošík.in
```

U programů v jazyce C++ je nejprve zkompilejte (např. pomocí `g++ -g -O2 -std=gnu++23 -static boucník.cpp -o boucník`) a poté spusťte:

```
python3 testing_tool.py ./boucník < hrošík.in
```

Ukázkový příklad

V prvním ukázkovém příkladu je skrytá permutace $[p_0, p_1, p_2, p_3, p_4] = [2, 1, 0, 3, 4]$. To splňuje omezení sad testů 2, 5 a 6. Nejprve program přečte celé číslo $N = 5$. Poté program požádá o jízdu s $K = 2$ zapnutými spínači: spínač 4 a spínač 0. Tyto spínače ovládají místnosti $p_4 = 4$ a $p_0 = 2$; viz obrázek níže. Erika slyší 3 výkřiky (označených šipkami na obrázku): poprvé, když atrakce projíždí z neosvětlené místnosti 1 do osvětlené místnosti 2; podruhé z osvětlené místnosti 2 do neosvětlené místnosti 3; a potřetí při přechodu z neosvětlené místnosti 3 do osvětlené místnosti 4. Program poté požádá o další jízdu, kde jsou místnosti p_0, p_2 a p_3 osvětlené, při které Erika uslyší 3 výkřiky. Nakonec program odpoví $A = 2$ a $B = 4$, což je skutečně správné, protože tyto odpovědi ovládají první a poslední místnost ($p_2 = 0$ a $p_4 = 4$). Všimněte si, že $A = 4$ a $B = 2$ by také byla správná odpověď.



Ve druhém ukázkovém příkladu je skrytá permutace $[p_0, p_1, p_2] = [2, 0, 1]$. To splňuje omezení sad 1, 2, 5 a 6. Program vyzkouší jízdu, při které jsou všechny tři přepínače zapnuté. Protože to znamená, že všechny místnosti jsou osvětlené, Erika neuslyší žádný výkřik. Při druhé jízdě se zapnou přepínače 1 a 0, takže místnosti $p_1 = 0$ a $p_0 = 2$ budou osvětlené, zatímco místnost 1 je neosvětlená. Erika slyší dva výkřiky: když jízda jede z místnosti 0 (osvětlená) do místnosti 1 (neosvětlená) a z místnosti 1 (neosvětlená) do místnosti 2 (osvětlená). Při poslední jízdě se nezapnou žádné přepínače, což znamená, že všechny tři místnosti jsou neosvětlené a Erika opět

neuslyší žádný výkřik. Program poté odpoví přepínači 1 a 0, které skutečně ovládají první a poslední místnost. Jak „ ! 0 1 “, tak „ ! 1 0 “ jsou akceptované odpovědi.

Ve třetím ukázkovém příkladu je skrytá permutace $[p_0, p_1, p_2, p_3] = [0, 1, 2, 3]$. To splňuje omezení sad 2, 3, 4, 5 a 6.

První ukázkový příklad

výstup hodnotiče	váš výstup
5	
	? 10001
3	
	? 10110
3	
	! 2 4

Druhý ukázkový příklad

výstup hodnotiče	váš výstup
3	
	? 111
0	
	? 110
2	
	? 000
0	
	! 1 0

Třetí ukázkový příklad

výstup hodnotiče	váš výstup
4	
	? 1010
3	
	! 0 3