Var är Waldo?

Problemnamn	whereswaldo
Tidsgräns	11 sekunder
Minnesgräns	1 gigabyte

Det finns en dold permutation $P_0, P_1, ..., P_{N-1}$ med längden N, som garanterat genereras likformigt slumpmässigt. Permutationen innehåller talen $1, 2, 3, ..., N$ exakt en gång vardera, i en okänd ordning.

Du kan välja positionerna l och r och ställa frågor av slaget: "Vad är summan av $P_l + P_{l+1} + \cdots + P_r$?"

Din uppgift är att hitta vilket P som är lika med 1 på så få frågor som möjligt. Du kommer få poäng utifrån hur många frågor du behöver.

Interaktion

Ditt program ska först läsa två heltal på på samma rad, T och N. T är antalet rundor som ditt program kommer att testas på, och N är längden på P.

Efter detta kommer T rundor:

När en runda börjar kan du börja ställa frågor. Skriva ut en rad med "`? a b`" för att fråga om summan av talen mellan positionerna a och b inklusive ($0 \leq a \leq b \leq N - 1$).

Efter varje fråga ska ditt program läsa ett heltal, summan av siffrorna i intervallet.

När du har hittat positionen för 1, skriv ut en rad med formen "`! i`", där i är indexet så att $P_i = 1$. När du har skrivit ut detta börjar nästa omgång.

Se till att använda flush för standardutdata efter att ha ställt en fråga, annars kan ditt program bedömas som Time Limited Exceeded (överskriden tidsgräns). I Python görs detta automatiskt med `print()`. I C++ så kommer `cout << endl;` att använda flush utöver att skriva ut en ny rad och om du använder printf, använd `fflush(stdout)`.

Begränsningar och poängsättning

Ditt program kommer att testas mot **ett enda testfall, med** $N = T = 1000$ (testfallet har flera rundor). Permutationen i varje runda kommer garanterat att **genereras slumpmässigt**.

Om din lösning gissar fel i någon av omgångarna kommer ditt bidrag att bedömas som *Wrong Answer* (fel svar).

Annars kommer dina poäng att beräknas enligt följande:

$$\text{poäng} = \min\left(220 - \tfrac{M}{2500}, 100\right),$$

där M är antalet frågor som ditt program ställer totalt under alla T-omgångar.

Poängen kommer att avrundas till närmaste heltal. Om poängen skulle bli negativa så kommer det behandlas som noll poäng.

Således, om du använder fler än $550\,000$ frågor kommer du att få 0 poäng, och om du använder $300\,000$ eller färre frågor får du 100 poäng. Däremellan växer dina poäng linjärt.

Testverktyg

För att underlätta testning av din lösning tillhandahåller vi ett enkelt verktyg som du kan ladda ner. Se "bilagor" längst ner på kattis problemsida. Verktyget är valfritt att använda, och du får ändra i koden. Observera att det officiella betygsprogrammet på kattis skiljer sig från testverktyget.

Exempel på användning (med T=1000, N=10):

För python-program, säg `solution.py` (körs normalt som `pypy3 solution.py`):

```
python3 testing_tool.py pypy3 solution.py <<<"1000 10"
```

För C++-program, kompilera det först (t.ex. med `g++ -std=gnu++17 solution.cpp -o solution.out`) och kör sedan:

```
python3 testing_tool.py ./solution.out <<<"1000 10"
```

Exempel

I provfallet nedan är $T = 2$ och $N = 10$. För den första av dessa två omgångar, säg att den dolda permutationen är "`6 10 8 7 9 1 2 4 5 3`". Den första frågan `? 0 9` frågar efter summan av alla tal, vilket är 55, och den andra frågan `? 0 4` ber om $6 + 10 + 8 + 7 + 9 = 40$.

testarens utdata	din utdata
2 10	
	? 0 9
55	
	? 0 4
40	
	? 5 5
1	
	! 5
	? 0 0
1	
	! 0