

Waar is Waldo?

Problem Name	whereswaldo
Time Limit	11 seconds
Memory Limit	1 gigabyte

Er is een verborgen permutatie P_0, P_1, \dots, P_{N-1} van lengte N , welke gegarandeerd uniform random gegenereerd is. The permutatie bevat de getallen $1, 2, 3, \dots, N$ exact 1 keer, in onbekende volgorde.

Je kan de posities l en r kiezen, en vragen stellen van de vorm: "Wat is de som van $P_l + P_{l+1} + \dots + P_r$?"

Jouw taak is het om de positie van de 1 in P te vinden met zo min mogelijk vragen. Je score hangt af van het aantal vragen dat je gebruikt.

Interactie

Jouw programma leest eerst 2 integers van de eerste regel, T en N . T is het aantal rondes waarop jouw programma wordt getest en N is de lengte van P .

Hierna komen T rondes:

Wanneer een ronde begint mag je vragen stellen. Print een lijn met "`? a b`" om de som op te vragen van de getallen tussen de posities a en b inclusief ($0 \leq a \leq b \leq N - 1$).

Na elke vraag leest je programma een integer, de som van de getallen in het interval.

Zodra je de positie van de 1 gevonden hebt, print een regel in de vorm "`! i`", waar i is de index zodat $P_i = 1$. Daarna begint de volgende ronde.

Flush the standaarduitvoer na iedere vraag, anders krijg je misschien een Time Limit Exceeded. In Python, `print()` flushes automatisch. In C++, `cout << endl;` flushed ook; als je `printf` gebruikt, gebruik `fflush(stdout)`.

Constraints and Scoring

Je programma wordt getest met **een enkele testcase met $N = T = 1000$** . De permutatie in elke test is gegarandeerd **random gegenereert**.

Als je oplossing fout raad in een ronde, dan geeft je hele oplossing *Wrong answer*.

Anders, wordt de score als volgt berekend:

$$\text{score} = \min\left(220 - \frac{M}{2500}, 100\right) \text{ points,}$$

where M is the number of questions your program asks in total over all T rounds.

The score wordt afgerond naar de dichtsbijzijnde integer. Als de score negatief wordt, dan krijg je nul punten.

Samenvattend, als je meer dan 550 000 vragen gebruikt krijg je 0 punten, en als je 300 000 of minder vragen gebruikt krijg je 100 punten. Daartussen groeit je score linear.

Testing Tool

Je kan een simpele tool downloaden om je oplossing lokaal te testen. Zie de bijlages onder aan de kattis probleem-pagina. Je hoeft de tool niet te gebruiken, en je mag hem aanpassen. Het officiële grading programma op kattis verschilt van de test-tool.

Voorbeeldgebruik (met $T=1000$, $N=10$):

In python programmas, zeg `solution.py` (normaal run als `python3 solution.py`):

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

In C++, compileer het eerst (e.g. with `g++ -std=gnu++17 solution.cpp -o solution.out`) en run daarna:

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

Voorbeeld

In het voorbeeld is $T = 2$ en $N = 10$. Voor de eerste ronde van de twee rondes, de verborgen permutatie is "6 10 8 7 9 1 2 4 5 3". De eerste vraag ? 0 9 krijgt de som van alle getallen terug en die is inderdaad 55, de tweede vraag ? 0 4 heeft als antwoord $6 + 10 + 8 + 7 + 9 = 40$.

grader output	your output
2 10	
	? 0 9
55	
	? 0 4
40	
	? 5 5
1	
	! 5
	? 0 0
1	
	! 0