

## Gdje je Valdo?

Problem Name	whereswaldo
Time Limit	11 sekunda
Memory Limit	1 gigabajt

Postoji skrivena permutacija  $P_0, P_1, \dots, P_{N-1}$  duljine  $N$ , koja je zajamčeno generirana ravnomjerno nasumično. Permutacija sadrži brojeve  $1, 2, 3, \dots, N$  točno jednom, u nekom nepoznatom redosljedu.

Možete odabrati pozicije  $l$  i  $r$  i postavljati pitanja oblika: "Koja je suma  $P_l + P_{l+1} + \dots + P_r$ ?"

Vaš zadatak je pronaći poziciju broja 1 u  $P$  koristeći što manje pitanja. Bodovat će se prema broju postavljenih pitanja.

## Interakcija

Vaš program prvo treba učitati dva cijela broja u jednom retku,  $T$  i  $N$ .  $T$  je broj rundi u kojima će se vaš program testirati, a  $N$  je duljina  $P$ .

Nakon toga slijedi  $T$  rundi:

Kada započne runda, možete postavljati pitanja. Ispisujte liniju "`? a b`" kako biste pitali za zbroj brojeva između pozicija  $a$  i  $b$  uključivo ( $0 \leq a \leq b \leq N - 1$ ).

Nakon svakog pitanja, vaš program treba učitati cijeli broj, zbroj brojeva u intervalu.

Kada pronađete poziciju broja 1, ispišite liniju oblika "`! i`", gdje je  $i$  indeks takav da  $P_i = 1$ . Nakon što ispišete to, započet će sljedeća runda.

Obavezno ispraznite standardni izlaz nakon postavljanja pitanja, inače vaš program može biti ocijenjen kao "Time Limit Exceeded". U Pythonu, `print()` automatski isprazni izlaz. U C++-u, `cout << endl;` također isprazni izlaz uz ispis novog retka; ako koristite `printf`, onda koristite `fflush(stdout)`.

## Ograničenja i bodovanje

Vaš program će biti testiran samo na jednom testnom slučaju, s  $N = T = 1000$ . Permutacija u svakom testu zajamčeno je generirana nasumično.

Ako vaše rješenje krivo pogodi u bilo kojoj rundi, vaša će predaja biti ocijenjena kao *Wrong answer*.

Inače, bodovi će biti izračunati na sljedeći način:

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

gdje je  $M$  broj postavljenih pitanja vašeg programa u ukupno svim  $T$  rundama.

Bodovi će biti zaokruženi na najbliži cijeli broj. Ako bodovi postanu negativni, tretirat će se kao nula bodova.

Stoga, ako koristite više od 550 000 pitanja, dobit ćete 0 bodova, a ako koristite 300 000 ili manje pitanja, dobit ćete 100 bodova. Između toga, vaš će se rezultat linearno povećavati.

## Testni alat

Da bismo olakšali testiranje vašeg rješenja, pružamo jednostavan alat koji možete preuzeti. Pogledajte "attachments" na dnu stranice problema na Kattisu. Upotreba alata je opcionalna i možete ga mijenjati. Napomena: službeni grader program na Kattisu razlikuje se od testnog alata.

Primjer korištenja (za  $T = 1000$ ,  $N = 10$ ):

Za Python programe, recimo `solution.py` (obično pokrenuto kao `pypy3 solution.py`):

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

Za C++ programe, prvo ga trebate prevesti (npr. s `g++ -std=gnu++17 solution.cpp -o solution.out`), a zatim pokrenuti s:

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

## Primjer

U primjeru testnog slučaja,  $T = 2$  i  $N = 10$ . Za prvu od te dvije runde, pretpostavimo da je skrivena permutacija "6 10 8 7 9 1 2 4 5 3". Prvo pitanje ? 0 9 traži zbroj svih brojeva, što je doista 55, a drugo pitanje ? 0 4 traži  $6 + 10 + 8 + 7 + 9 = 40$ .

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