

Kopač Samuel

Ovo je interaktivni zadatak

Zadatak

Drevno kraljevstvo grada Salamisa krije blago neprocjenjive vrijednosti zakopano na dubini n metara ispod površine. Vaš zadatak je naći broj n pomoću eksperimenata koristeći vašeg robota za kopanje, Samuela.

Za svaki eksperiment robotu date niz cijelih brojeva a_1, a_2, \dots, a_k , nakon čega će on prokopati tačno do dubine n koristeći samo korake veličine a_i . Ovo znači da, ako je robot trenutno na dubini x , on može prokopati na dubinu $x + a_1, x + a_2, \dots$, ili $x + a_k$. Kada robot dođe na dubinu tačno n on će vam iskomunicirati broj koraka koji je napravio. Robot je uvijek efikasan i pravi najmanji mogući broj koraka potreban da dostigne dubinu n .

Primjera radi, neka je skriveni broj $n = 23$. Ako robotu date niz $a = [1, 3, 8]$ on će dostići dubinu n u 5 koraka (na primjer, $0 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 15 \rightarrow 23$). Ako robotu date niz $a = [9, 5]$ on će dostići dubinu n u 3 koraka ($0 \rightarrow 9 \rightarrow 18 \rightarrow 23$).

Vaš zadatak je pogoditi broj n nakon pravljenja nekoliko eksperimenata sa sljedećim ograničenjima:

- Skriveni broj n nije manji od 1, niti veći od 30000.
- Ukupni broj eksperimenata ne smije biti veći od 20.
- Zbir veličina nizova a u svim eksperimentima ne smije biti veći od 250.
- U svakom eksperimentu mora biti moguće dostići dubinu n .

Interakcija

Kako bi se eksperiment pokrenuo vaš program treba ispisati jedan red u obliku: `? k a_1 a_2 \dots a_k`, gdje je k ($1 \leq k \leq 250$) veličina niza i a_i ($1 \leq a_i \leq 30000$) su elementi niza. Nakon toga, vaš program treba pročitati cijeli broj m , najmanji broj koraka potreban robotu da dostigne dubinu n .

Kada je vaš program spreman pogoditi broj n treba ispisati jedan red u obliku: `! n`. Vaš program nakon toga treba završiti sa izvršavanjem.

Nemojte zaboraviti "flushati" izlaz nakon svakog ispisa.

Ako, prilikom bilo kojeg koraka interakcije, pročitate -1 umjesto ispravnih podataka vaše rješenje mora odmah završiti sa radom. Ovo znači da će se vaše rješenje smatrati neispravnim zbog neispravnog eksperimerimenta ili bilo koje druge greške. Ako se rad programa ne obustavi to može dovesti do besmislene presude zato što vaš program nastavlja čitati iz zatvorenog toka.

Kako biste "flushali" izlaz, možete koristiti:

- `fflush(stdout)` ili `cout.flush()` u C++-u;
- `sys.stdout.flush()` u Pythonu;

Primjer

Ulaz	Izlaz
	? 3 1 3 8
5	
	? 2 9 5
3	
	! 23

Bodovanje

Vaše rješenje će biti testirano na 100 testnih primjera. Ako vaše rješenje ne uspije pogoditi broj n , ili prekorači zadana ograničenje u bilo kojem testnom primjeru dobit ćete 0 bodova.

Ako vaše rješenje uspješno pogodi broj n i ne prekorači ograničenja ni u jednom testnom primjeru broj bodova će biti određen na osnovu ukupnog broja koraka koje je robot napravio u svim eksperimentima.

Neka je a_i ukupan broj koraka koje je robot napravio u svim eksperimentima testa i , i neka je S maksimalna vrijednost s_i iz svih testnih primjera. Vaš broj bodova će biti izračunat iz sljedeće tabele:

S	Broj bodova
≤ 100	100
101 – 150	95
151 – 200	90
201 – 250	80
251 – 300	70
301 – 500	60
501 – 1000	50
1001 – 5000	20
5001 – 20000	10
20001 – 30000	5
≥ 30001	0