

# Digger

*Ovo je interaktivni zadatak*

## Zadatak

Tokom iskopavanja ruševina drevnog polisa Salamine, arheolozi su našli skriveni kraljevski grob, zakopan na nepoznatih  $n$  metara ispod zemlje. Vaš zadatak je da odredite vrednost broja  $n$  tako što ćete raditi eksperimente sa Vašim botom za kopanje.

U svakom eksperimentu dajete botu niz celih brojeva  $a_1, a_2, \dots, a_k$  i tražite od njega da kopa od površine (dubina 0) do dubine  $n$  koristeći samo korake dužine  $a_i$ . Znači, ako je robot trenutno na dubini  $x$ , može da pređe na dubine  $x + a_1, x + a_2, \dots, x + a_k$ . Kada bot dostigne dubinu  $n$ , kaže Vam broj koraka koji mu je bio potreban. Bot je veoma inteligentan i uvek iskoristi najmanji mogući broj koraka da dostigne dubinu  $n$ .

Na primer, neka je skriveni broj  $n = 23$ . Ako damo botu niz  $a = [1, 3, 8]$ , on stiže do dubine  $n$  u 5 koraka (na primer  $0 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 15 \rightarrow 23$ ). Ako mu damo niz  $a = [9, 5]$ , može da stigne do dubine  $n$  u 3 koraka ( $0 \rightarrow 9 \rightarrow 18 \rightarrow 23$ ).

Vaš zadatak je da nađete broj  $n$  nakon što izvršite više eksperimenata sa sledećim ograničenjima:

- Skriveni broj  $n$  je između 1 i 30000 (uključujući i njih).
- Ukupan broj eksperimenata treba da bude najviše 20.
- Veličina niza  $a$  u svim eksperimentima treba da bude najviše 250.
- U svakom eksperimentu treba da bude moguće da bot stigne do dubine  $n$ .

## Interakcija

Da biste izvršili eksperiment, Vaš program treba da ispiše jednu liniju u formatu  $? \ k \ a_1 \ a_2 \ \dots \ a_k$ , gde je  $k$  ( $1 \leq k \leq 250$ ) veličina niza, dok su  $a_i$  ( $1 \leq a_i \leq 30000$ ) elementi samog niza. Nakon toga, Vaš program treba da učitava jedan ceo broj  $m$ , najmanji broj koraka koji je bio potreban botu da stigne do dubine  $n$ .

Kada je Vaš program siguran u vrednost  $n$ , treba na izlazu da ispiše liniju:  $! \ n$ . Nakon toga program treba da se završi.

Ne zaboravite da nakon ispisa svakog eksperimenta uradite flush izlaza.

Ako u bilo kom koraku interakcije sa ulaza učitate `−1` umesto očekivanog broja, Vaš program mora da se odmah završi. To znači da će se Vaše rešenje smatrati netačnim zato što ste napravili nepravilan upit ili neku drugu grešku. Ukoliko se program ne završi, može Vam biti data pogrešna informacija o grešci, pošto će Vaš program nastaviti da čita iz zatvorenog toka podataka.

Da biste uradili flush izlaza možete koristiti:

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

---

## Primer

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

---

## Bodovanje

Vaše rešenje će biti testirano na 100 testova. Ako i u jednom testu Vaše rešenje ne uspe da odredi broj  $n$  ili pređe data ograničenja, dobićete 0 bodova.

Inače, ako Vaše rešenje tačno odredi broj  $n$  i ne prekorači ograničenja ni u jednom testu, vaš broj bodova će biti izračunat u zavisnosti od ukupnog broja koraka koje je bot napravio u svim eksperimentima.

Neka je  $s_i$  ukupan broj koraka koje je bot napravio u svim eksperimentima u testu  $i$ , i neka je  $S$  maksimum od  $s_i$  za svih 100 testova. Vaš broj poena se računa na osnovu sledeće tabele:

$S$	Broj bodova
$\leq 100$	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
$\geq 30001$	0