

# Копач

Това е интерактивна задача.

## Задача

По време на разкопките на руините на древнокипърския полис Саламис археолог откри скрита царска гробница, закопана на скрита дълбочина от  $n$  метра под повърхността. Вашата задача е да намерите скритото число  $n$ , изпълнявайки експерименти с Вашия копач-робот.

Във всеки експеримент Вие давате на робота Ви списък от цели числа  $a_1, a_2, \dots, a_k$  и той ще дълбае от повърхността (дълбочина 0) до дълбочина  $n$ , ползвайки само стъпки с дължини сред числата в списъка. Това означава, че ако в момента роботът е на дълбочина  $x$ , той може да се придвижи до дълбочини  $x + a_1, x + a_2, \dots$ , или  $x + a_k$ . Когато роботът достигне дълбочина  $n$ , той ти казва колко стъпки е направил. Роботът е много интелигентен и винаги прави минималния брой стъпки, нужни за да достигне дълбочина  $n$ .

Примерно, нека скритото число да е  $n = 23$ . Ако подадете списъка  $a = [1, 3, 8]$  на робота, той може да стигне  $n$  с 5 стъпки (на пример  $0 \rightarrow 3 \rightarrow 4 \rightarrow 7 \rightarrow 15 \rightarrow 23$ ), а ако дадете списъка  $a = [9, 5]$  на робота, той може да достигне  $n$  с 3 стъпки ( $0 \rightarrow 9 \rightarrow 18 \rightarrow 23$ ).

Вашата задача е да определите  $n$ , правейки няколко експеримента, удовлетворявайки следните ограничения:

- Скритото число  $n$  е в интервала 1 to 30 000 включително.
- Общият брой експерименти е най-много 20.
- Сумата от размерите на множествата  $a$  във всички експерименти трябва да е не повече от 250.
- Във всеки експеримент трябва да е възможно роботът да достигне дълбочина  $n$ .

## Комуникация

За да изпълни експеримент, Вашата програма трябва да изведе един ред от следния вид:  $? \ k \ a_1 \ a_2 \ \dots \ a_k$ , където  $k$  ( $1 \leq k \leq 250$ ) е размерът на списъка, а  $a_i$  ( $1 \leq a_i \leq 30000$ ) са елементите на списъка. След това Вашата програма трябва да прочете едно цяло число  $m$  - минималният брой стъпки, от които роботът се е нуждаел, за да достигне дълбочина  $n$ .

Когато Вашата програма е установила стойността на  $n$ , трябва да изведете единствен ред от вида: `! n`. След това програмата Ви трябва да приключи.

Не забравяйте да flush-нете изхода и да изведете нов ред след описанието на всеки експеримент.

Ако в някакъв момент в комуникацията прочетете `-1` вместо някакво валидно число, решението Ви трябва да се прекрати веднага. Това означава, че решението Ви ще получи **Wrong answer**, заради невалиден експеримент или друга грешка. Ако не прекратите програмата, е възможно да получите произволно съобщение за край на теста, понеже четете от затворен поток.

За да flush-нете, ползвайте:

- `fflush(stdout)` или `cout.flush()` в C++;
- `sys.stdout.flush()` в Python;

---

## Пример

Вход	Изход
	? 3 1 3 8
5	
	? 2 9 5
3	
	! 23

---

## Оценяване

Вашето решение ще бъде тествано на 100 теста. Ако решението Ви не успее да установи правилната стойност на  $n$  или превиши дадените ограничения на поне един тест, ще получите 0 точки за това решение.

Ако решението Ви успешно установи  $n$  и не превиши ограниченията във всеки тест, Вашият резултат ще се изчисли на база сумарния брой стъпки, които роботът Ви е извършил във всички експерименти.

Нека  $s_i$  да бъде общия брой стъпки, ползвани във всеки експеримент от тест  $i$ , и нека  $S$  да бъде стойността на максималното  $s_i$  сред всички 100 теста. Тогава Вашият резултат ще бъде пресметнат въз основа на следната таблица:

$S$	Точки
$\leq 100$	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