

Plan

Limit pamięci: 256 MB

"A teraz prędko, zanim dotrze do nas, że to bez sensu."

~ Król Julian, Madagaskar 2

Zespół Króla Juliana spieszy się, by zrobić jak najwięcej, nim dotrze do nich informacja, że cały plan jest bez sensu. Niewykluczone jednak, że pewnego dnia ta wiadomość do nich dotrze. To będzie chwila, gdy zapał zespołu spadnie do zera lub będzie ujemny.

Dasz radę stwierdzić kiedy to nastąpi? Znasz początkowy zapał całego zespołu oraz jego zmiany po każdej akcji.

Wejście

W pierwszym wierszu znajdują się oddzielone spacją trzy liczby całkowite n , k oraz s oznaczające odpowiednio:

n ($1 \leq n \leq 10^6$) – liczbę akcji bezsensownego planu,

k ($1 \leq k \leq 10^{16}$) – liczbę powtórzeń planu,

s ($-10^{17} \leq s \leq 10^{17}$) – początkowy zapał zespołu.

W drugim wierszu znajduje się ciąg n liczb całkowitych $a_1, a_2, \dots, a_n$ ($-2 \cdot 10^{18} \leq a_i \leq 2 \cdot 10^{18}$), gdzie a_i to wartość, o którą zmienia się zapał po wykonaniu i -tej akcji planu. Dodatkowo wartość bezwzględna z sumy wartości na każdym prefiksie nie przekracza $2 \cdot 10^{18}$. Formalnie: $\forall 1 \leq i \leq n \mid \sum_{j=1}^i a_j \mid \leq 2 \cdot 10^{18}$.

Wyjście

Zespół króla Juliana orientuje się, że plan jest bez sensu, jeśli jego zapał osiąga zero lub liczbę ujemną. Zespół powinien wykonać plan k razy. Twój program powinien wypisać:

- A. Jeśli od razu na początku zapał jest zero lub ujemny ($s \leq 0$), Twój program powinien wypisać: `Julian zgasl...`
- B. Numer akcji (wartość od 1 do n), po której zapał staje się zerem lub ujemny, czyli dociera do zespołu, że plan i jego powtarzanie jest bez sensu.
- C. Jeśli w trakcie oraz po wykonaniu k razy planu zapał zawsze był większy od zera, Twój program powinien wypisać:
`Nic nas nie powstrzyma!`

Przykłady

Wejście dla testu `r4b0a`:

```
4 1000 -2
5 -1 8 0
```

Wyjście dla testu `r4b0a`:

```
Julian zgasl...
```

Wyjaśnienie: Zespół od razu – przed wykonaniem planu – zorientował się, że powtarzanie planu bez sensu. Początkowy zapał wynosi -2 , czyli jest mniejszy lub równy 0. Wypisujemy: `Julian zgasl...`

Wejście dla testu `r4b0b`:

```
4 2000 9
-2 2 -7 1
```

Wyjście dla testu `r4b0b`:

```
3
```

Wyjaśnienie: Plan zawiera 4 akcje. Zespół powinien powtórzyć plan 2000 razy. Początkowy zapał wynosi 9.

Plan

Limit pamięci: 256 MB

Iteracja	Akcja	Zmiana zapału	Zapał po akcji
1	1	-2	7
1	2	+2	9
1	3	-7	2
1	4	+1	3
2	1	-2	1
2	2	+2	3
2	3	-7	-4

Po akcji 3 w drugiej iteracji otrzymaliśmy zapał równy zero lub mniejszy od zera. Przerywamy wszystko. Przerywamy kolejne akcje i kolejne iteracje planu choć jesteśmy dopiero na iteracji planu numer 2 spośród 2000. Wypisujemy **numer** akcji, kiedy zapał spadł do zera lub mniej: 3

Wejście dla testu r4b0c:

```
2 3 1
5 -4
```

Wyjście dla testu r4b0c:

```
Nic nas nie powstrzyma!
```

Wyjaśnienie: Plan zawiera 2 akcje. Zespół powinien powtórzyć plan 3 razy. Początkowy zapał wynosi 1.

Iteracja	Akcja	Zmiana zapału	Zapał po akcji
1	1	+5	6
1	2	-4	2
2	1	+5	7
2	2	-4	3
3	1	+5	8
3	2	-4	4

Zapał nigdy nie spada do zera ani poniżej. Wypisujemy: Nic nas nie powstrzyma!

Pozostałe testy przykładowe

- test r4b0d: $n = 2026, k = 1, s = 1234$. Wartości a_i wynoszą $-1, -2, -3, -4, \dots, -n$. Wynik to 50.
- test r4b0e: $n = 1\,000\,000, k = 2026, s = 10^{17}$. Wartości a_i wynoszą $-1, -1, -1, -1, \dots, -1$. Zapał zawsze dodatni.

Ocenianie

Podzadanie	Ograniczenia	Limit czasu	Punkty
1	$n = 1, k = 1, 0 \leq s \leq 10^6$	3 s (C++) / 7 s (Python)	10
2	$n = 1, k = 1$	3 s (C++) / 7 s (Python)	10
3	$n \leq 1000, k = 1$	3 s (C++) / 7 s (Python)	20
4	$k = 1$	3 s (C++) / 7 s (Python)	20
5	Brak dodatkowych ograniczeń	3 s (C++) / 7 s (Python)	40