

## Име на задачата: Топки

Language: Bulgarian

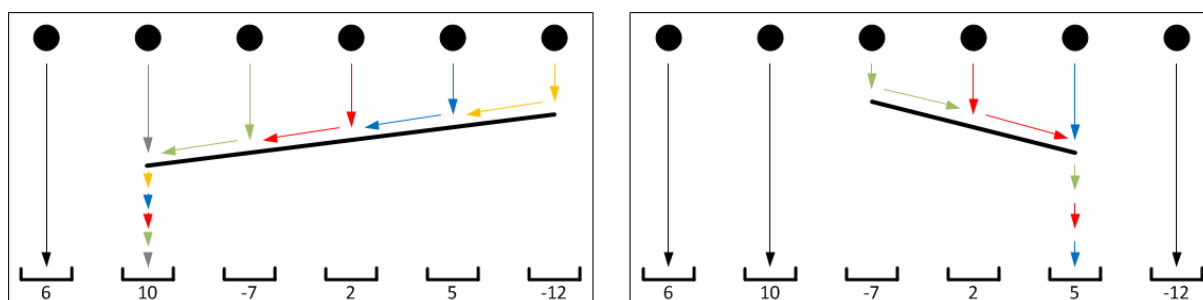
Source code: balls.c, balls.cpp, balls.pas  
Input / Output files: balls.in / balls.out  
Time limit: 0.2 s  
Memory limit: 256 MB

### Описание на задачата

Дадена е стандартна координатна система, както и  $n$  топки и  $n$  дупки в нея: топките се намират на позиции  $(1, h), (2, h), \dots, (n, h)$  за някакво  $h > 0$ , а дупките се намират на позиции  $(1, 0), (2, 0), \dots, (n, 0)$ . В даден момент всички топки започват да падат (успоредно на ординатната ос, в отрицателна посока). За всяка топка, която падне в  $i$ -тата дупка, получаваме  $c_i$  точки. Разбира се, за да се избегне тривиалния резултат  $c_1 + c_2 + \dots + c_n$ , е разрешено да се поставя препятствие.

**Ляво препятствие** е отсечка  $(x_1, y_1) - (x_2, y_2)$ , за която:  $x_1, x_2 \in \mathbb{N}, 1 \leq x_1 < x_2 \leq n$  и  $0 < y_1 < y_2 < h$  (т.е. препятствието е наклонено *наляво*). **Дясно препятствие** е отсечка  $(x_1, y_1) - (x_2, y_2)$ , за която  $x_1, x_2 \in \mathbb{N}, 1 \leq x_1 < x_2 \leq n$  и  $h > y_1 > y_2 > 0$  (т.е. препятствието е наклонено *надясно*). Ако топката срещне препятствие падайки, тя временно спира, след това се плъзга надолу по препятствието и накрая пада направо в дупката под долния край на препятствието.

Лявото изображение показва траекториите на топките, когато срещнат ляво препятствие, докато на дясното изображение може да видим траекториите на топките, когато срещнат дясно препятствие. Забележете, че когато има препятствие, има дупки, в които попадат няколко топки и дупки, в които няма топки; **обаче всяка топка пада в една от дупките.**



Броят на точките за всяка от дупките е известен. Вашата задача е да отговорите на следните два въпроса:

- Какъв е най-високият резултат, който може да постигнете, ако трябва да поставите точно едно дясно препятствие?
- Какъв е най-високият резултат, който може да постигнете, ако трябва да поставите точно едно ляво препятствие?

### Вход

На първия ред на входа има едно цяло число  $n$  – броя на топките и на дупките. На втория ред на входа, разделени с интервали, са зададени  $n$ -те цели числа  $c_i$  – броят на точките за всяка от дупките, подредени отляво надясно.

### Изход

На първия ред на изхода трябва да е записано едно цяло число – отговорът на първия въпрос. На втория ред на изхода трябва да е записано едно цяло число – отговорът на втория въпрос. **Трябва да използвате 64-битов целочислен тип данни.**

### Пример

Вход:	Изход:
6	19
6 10 -7 2 5 -12	56

### Пояснение на примера

Оптималното поставяне на дясно/ляво препятствие е показано на изображенията. При поставянето на дясно препятствие както е показано на дясното изображение, топки 3, 4 и 5 ще попаднат в дупка номер 5 и резултатът ще бъде  $6 + 10 + 5 + 5 + 5 + (-12) = 19$  точки. Аналогично, при поставяне на ляво препятствие както е показано на лявото изображение, ще получим  $6 + 10 + 10 + 10 + 10 + 10 = 56$  точки. Не е възможно да се получат повече точки при друго разположение на препятствията.

### Ограничения

- $3 \leq n \leq 300\,000$ .
- $-10^9 \leq c_i \leq 10^9$ .
- В 30% от тестовите примери  $|c_i| \leq 100$ , за всяко  $1 \leq i \leq n$ .
- Ако дадете верен отговор **само на един** от въпросите, вие ще получите 20% от точките за тестовия пример, но вие трябва винаги да извеждате по **две** числа.