





# ЕСЕНЕН ТУРНИР ПО ИНФОРМАТИКА

## София, 24 – 26 ноември 2023 г.

### Група А, 11 – 12 клас

 : 4 сек.  
 : 600 MB

#### Задача А1. БОГАТО МОМИЧЕ

Мария мечтае един ден да бъде богато момиче. За да сбъдне нейното желание, Автора я поставя на пътя към успеха - редица  $A$  от  $N$  ненулеви числа. Парите обаче идват след много труд и Автора няма да остави Мария да заобиколи тази капиталистическа догма.

Мария може да изпълнява следното действие: ако в началото тя е на позиция  $pos$ , може да "скочи" на позиция  $pos + k$ ,  $k > 1$ . След като направи няколко скока с еднаква дължина  $k$  (ще означим броя им с  $t$ ) и съответно стигне на позиция  $new$ , тя ще седне да почива и ще получи  $t \times A_{new}$  лева. Забележете, че тя може да направи няколко скока (да кажем 5) с еднаква дължина и да си почине някъде по средата при изпълнението им (да кажем след втория). Тогава тя ще получи отделно пари за двете си почивки (да кажем  $2 \times A_i + 3 \times A_j$ , а не  $5 \times A_j$ ).

Вие не се интересувате от Мария, но ситуацията Ви впечатлява от оптимизационна гледна точка. Намерете максималния брой левове, които Мария може да получи, ако започва от позиция 1 и завършва на позиция  $N$ .

#### Задача

Напишете програмата `jumps`, съдържаща функция `max_prize`, която ще се компилира с програмата на журито и успешно ще решава гореописания проблем.

#### Детайли по имплементацията

Функцията `long long max_prize(std::vector<int> A)`, която трябва да напишете, ще бъде извикана само веднъж от програмата на журито и като аргумент ще получи редицата  $A$ .

Вашата програма `jumps.cpp` трябва да имплементира функцията `max_prize`. Тя може да съдържа и друг код, и функции, необходими за работата Ви, но не трябва да съдържа главната функция `main`. Също така, не трябва да четете от стандартния вход или да отпечатвате на стандартния изход. Програмата Ви трябва да включва хедър файла `jumps.h` чрез указание към компилатора:

```
#include "jumps.h"
```

#### Ограничения



- $3 \leq N \leq 2 \times 10^5$
- $1 \leq |A_i| \leq 10^9$

#### Пример

Вход	Изход	Обяснение на примера
5 1 2 -1 3 100	200	Оптималният отговор получаваме, когато скочим два пъти през позиции 1 – 3 – 5 и получим $2 \times A_5 = 2 \times 100$ . Напълно валидно е също да скочим на позиция 3, да загубим един лев и на позиция 5 да получим 100 - общо 99 лева. Това обаче е субоптимално. Не можем да скочим през позиции 1 – 2 – 3 – 4 – 5, защото всеки скок е с дължина 1, а това е забранено.



**ЕСЕНЕН ТУРНИР ПО ИНФОРМАТИКА**  
**София, 24 – 26 ноември 2023 г.**  
**Група А, 11 – 12 клас**

 : 4 сек.  
 : 600 MB

Подзадачи

Подзадача	Точки	$N$	Други ограничения	Необходими подзадачи
1	0	–	Примерният тест.	–
2	3	$\leq 20$	–	–
3	10	$\leq 3 \times 10^2$	–	2
4	7	$\leq 2 \times 10^3$	–	2 – 3
5	15	$\leq 3 \times 10^4$	–	2 – 4
6	18	$\leq 2 \times 10^5$	$0 < A_i$ за всяко $i$	–
7	18	$\leq 1 \times 10^5$	–	2 – 5
8	29	$\leq 2 \times 10^5$	–	1 – 7

Точките за дадена подзадача се получават само ако се преминат успешно всички тестове, предвидени за нея.

Локално тестване

Предоставен Ви е файлът `Lgrader.cpp`, който може да компилирате заедно с вашата програма, за да я тествате. При стартиране програмата ще чете от стандартния вход стойността на  $N$ , след което ще чете числата  $A_i$ . След това ще се отпечата комуникацията, която се извършва. Може да модифицирате предоставения файл, както искате.