



### за разширения национален отбор

### Група А (старша възраст)

1. август 2022 г.

## ЗАДАЧА АТ6. ХАЗНА

Очакваше се да е тиха нощ. Павел седеше зад мониторите, охранявайки сградата. Той работеше за Байтгарската централна банка. Работата му не беше особено интересна, но се заплащаше добре, заради което Павел не се оплакваше от нея. Изведнъж той забеляза, че сигналът от една от камерите беше изчезнал. Реши да провери. Като стигна до там, видя пламъци, излизаци от една от стаите.

– Трябва да проверя трезора... – помисли си той и изтича до горния етаж, където бяха парите. – Трябва да бъдат преместени на сигурно място – измърмори под носа си, но се усети, че не може просто да пхне парите в чувал и да ги премести...

Парите са сложени на  $N$  купчини. Всяка купчина съдържа определен брой пликосе с банкноти. Една от купчините обаче е различна – съдържа фалшиви банкноти, пълни с мастило, които се взривяват, когато напуснат трезора. Това е една от известните мерки за безопасност на Байтгарската централна банка. Фалшивият плик банкноти е малко по-тежък от истинския – тежи 101 грама за разлика от истинския, който тежи 100 грама. Само една от купчините съдържа фалшиви банкноти. Тази купчина не съдържа пликосе с истински банкноти.

В ъгъла на трезора има много точна електрическа везна, която позволява намирането на тежестта на произволно множество от пликосе. За жалост тя работи много бавно, а няма много време. Напишете програма **treasury**, която комуникира с библиотека и намира купчината с фалшиви банкноти, правейки минимален брой претегляния.

### Комуникация

За да използвате библиотеката, въведете в началото на програмата:

```
#include "skalib.h"
```

Библиотеката съдържа следните функции:

- `vector<int> inicjuj();`  
Тази функция връща информация за съдържанието на трезора. Тя връща вектор, съдържащ  $N$  цели числа  $A_1, A_2, \dots, A_N$ , като числото  $A_i$  е броят на банкнотите.
- `long long waz(const vector<int>& p);`  
Тази функция извършва претеглянето. Аргументът ѝ е вектор, съдържащ точно  $N$  цели числа  $P_1, P_2, \dots, P_N$ , като, разбира се, трябва да е изпълнено  $0 \leq P_i \leq A_i$  за всяко  $1 \leq i \leq N$ . Числото  $P_i$  показва броя на пликосе от  $i$ -тата купчина, които Павел е сложил на везната. Тази функция връща теглото на парите в грамове.
- `void odpowiedz(int k)`  
Тази функция казва на библиотеката номера на купчината, която съдържа фалшивите пликосе с банкноти. Извикването ѝ спира изпълнението на програмата.

Програмата Ви трябва да съдържа функция **main()**, но не трябва да чете нито от стандартния вход, нито от други файлове. Не може също да пише нито в файлове, нито към стандартния изход. Може да пише към стандартна грешка, но имайте предвид, че това консумира от ценното време за изпълнение.

# ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ ПО ИНФОРМАТИКА

„Д-р Младен Манев“

за разширения национален отбор

Група А (старша възраст)

1. август 2022 г.

## Ограничения

$$1 \leq N \leq 1\,000\,000;$$

$$1 \leq A_i \leq 10^9$$

## Примерна комуникация

Таблицата по-долу показва примерна комуникация с библиотеката:

Извикана функция	Резултат	Обяснение
<code>inicjuj()</code>	[1, 2]	$N = 2, A_1 = 1, A_2 = 2$
<code>waz([1, 0])</code>	101	$P_1 = 1, P_2 = 0$ - Павел поставя един плик от първата купчина на везната. Теглото на плика е 101 грама (ако втората купчина съдържаше фалшиви пликове, щеше да е 100 грама).
<code>odpowiedz(1)</code>	-	Даваме отговор и излизаме от програмата.

## Тестове за „оценка“

- $N = 3$ , всички купчини са с размер 1 с изключение на една. Едно претегляне е достатъчно;
- $N = 10$ , всички купчини са с размер 1. Четири претегляния са достатъчни;
- $N = 1000$ , купчините са с размер най-много 1000. Две претегляния са достатъчни;
- $N = 1\,000\,000$ , купчините са с размер най-много 1000. Десет претегляния са достатъчни.

## Оценяване

Нека с  $W$  да отбележим минималния брой претегляния, нужни, за да намерим фалшивата купчина. Ако програмата Ви намери коректно купчината, използвайки най-много  $W$  извиквания на функцията `waz()`, то тогава тя ще получи 100% от точките за теста. Ако извика функцията `waz()`  $W + 1$  пъти – 50%, а ако са  $W + 2$  – 25%. Ако програмата Ви не намери правилно фалшивата купчина или извика функцията `waz()` поне  $W + 3$  пъти, то тогава получавате 0 точки за теста.

Подзадача	Точки	Ограничения
1	8	Всички купчини са с размер 1
2	8	Най-много едно претегляне е нужно ( $W \leq 1$ )
3	28	$N \leq 1000$ ; най-много две претегляния са нужни ( $W \leq 2$ )
4	12	Всички купчини са с един и същи размер
5	32	$N \leq 100\,000$
6	12	Без допълнителни ограничения

Точките за всяка подзадача се получават при успешно преминаване на всички тестове, предвидени за нея.

# ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ ПО ИНФОРМАТИКА

„Д-р Младен Манев“

за разширения национален отбор

Група А (старша възраст)

1. август 2022 г.

## Локално тестване

Предоставени са Ви хедър файла `skalib.h`, както и локален грейдър `Lgrader.cpp`, които може да компилирате заедно с програмата си, за да я тествате. При стартиране на получения изпълним файл програмата ще очаква да въведете стойност на  $N$  – броя на купчините с пари, както и такава на  $K$ , която е между 1 и  $N$  и показва номера на купчината, съдържаща фалшиви пликосе с пари. На следващия ред е необходимо да въведете стойностите на редицата  $A_1, A_2, \dots, A_N$ , задаваща броя на пликосете с пари във всяка от купчините. Имате право да променяте предоставените Ви файлове, както намерите за добре.