

## Задача 2. TWINS

🕒 0.5 сек. 📄 256 MB

За пореден път близначките Рени и Нора успяха да заблудят всички, разменяйки местата си. Алекс беше убеден, че е измислил сигурна стратегия как да ги различава. Но се оказва, че е сгрешил. Рени и Нора, знаейки че той е безнадежно объркан, решили да изпробват върху него една от новоизмислените от тях игри. Тя се състояла в следното:

Избират  $N$  на брой различни свои снимки, номерирани с числата от 1 до  $N$ , една или две от които са на Рени, а останалите на Нора. Алекс има право да задава въпроси от вида: „Измежду снимките с номера  $p_1, p_2, \dots, p_K$  ( $1 \leq K \leq N$ ) има ли поне една, която да е на Рени?“. За да е по-интересно обаче, той има право да зададе цяла група с няколко такива въпроса и да получи едновременно отговорите за всеки от тях.

Целта на играта е Алекс да познае кои снимки са на Рени и кои на Нора след някакъв краен брой въпроси (което само по себе си би било постижение за него), но също така и да минимизира както броя групи с въпроси, така и общия брой въпроси в тях. За да се справи обаче с всичко това, той има нужда от Вашата помощ. Напишете програма `twins`, която да открива търсените една или две снимки.

### Задача

Напишете програма `twins`, която по данните за снимките и отговорите, да определя кои от тях са на Рени и кои на Нора. Тя трябва да съдържа функцията `play`, която ще се компилира с програмата на журито.

### Детайли по имплементацията

Функцията `play` трябва да има следния формат:

```
std::vector<int> play (int N, int T);
```

Тя се вика веднъж в началото от програмата на журито с два параметъра: броя на снимките  $N$  и колко от тях са на Рени  $T$  ( $T \in \{1, 2\}$ ). Функцията трябва да върне номерата на търсените една или две снимки. Ако снимките са две, редът на двете числа няма значение.

За комуникация с програмата на журито Ви се предоставя следната функция:

```
std::string guess (std::vector< std::vector<int> >& questions);
```

Като параметър `q` се подава група от въпроси, всеки съдържащ номера на снимки. Например, ако Алекс иска да зададе 2 въпроса и първият е за 1-вата и 2-рата снимка, а вторият за 2-рата и 3-тата, като параметър трябва да `q` се подаде `{ {1, 2}, {2, 3} }`. Позволено е да има "празни" въпроси, които да не питат за нищо (да подадете `{ { }, {2, 3} }` като параметър), като те **няма да се зачитат** към общия брой използвани въпроси.

Като отговор тя връща низ съставен от 0 и 1, указващи съответно отговори "не" и "да" на въпросите. Нека сте подали  $M$  въпроса в група. Те са номерирани с числата от 0 до  $M - 1$  в реда, в който сте ги подали, и на тях отговарят съответните индекси в низа.

Вие трябва да предадете към системата файл `twins.cpp`, който съдържа функцията `play`. Той може да съдържа и друг код, функции, както и глобални променливи 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 `twins.h` чрез указание към предпроцесора:

```
#include "twins.h"
```

### Оценяване

Всеки тест получава оценка, която е дробно число между 0 и 1 включително. Ако един тест има положителна оценка, то той се счита за успешен тест за вашето решение. Тъй като Нора и Рени поощряват дори и неоптималните игри, те използват следната система за оценяване.

Нека  $C$  е броят извиквания на функцията `guess`. Оценяването спрямо  $C$  за **всички тестове** е следното:

Процент $P_C$	$C$
100	$= 1$
60	$\leq 3$
15	$\leq 17$
10	$\leq 34$
0	$> 34$

Нека  $L$  е сумарният брой въпроси, които сте използвали в рамките на всички извиквания на функцията `guess`.

### Оценяване при $T = 1$

При  $T = 1$  има два вида тестове. В първия  $N \in [15, 30]$  (първата таблица), а във втория  $N = 50\,000$  (втората таблица). Оценяването спрямо  $L$  в тестовете с  $T = 1$  е следното:

Процент $P_L$	$L$
100	$\leq 10$
10	$\leq 30$
0	$> 30$

Процент $P_L$	$L$
100	$\leq 16$
$100 - \frac{L-16}{50-16} \times 20$	$\leq 50$
$80 - \left(\frac{L-50}{2000-50}\right)^{0.4} \times 35$	$\leq 2000$
$45 - \left(\frac{L-2000}{50000-2000}\right)^{0.3} \times 40$	$\leq 50\,000$
0	$> 50\,000$

### Оценяване при $T = 2$

При  $T = 2$  във всички тестове  $N = 50\,000$ . Оценяването спрямо  $L$  в тестовете с  $T = 2$  е следното:

Процент $P_L$	$L$
100	$\leq 152$
$100 - \left(\frac{L-152}{260-152}\right)^{0.5} \times 15$	$\leq 260$
$85 - \frac{L-260}{290-260} \times 10$	$\leq 290$
$75 - \left(\frac{L-290}{550-290}\right)^{0.3} \times 10$	$\leq 550$
$65 - \left(\frac{L-550}{2000-550}\right)^{0.6} \times 20$	$\leq 2000$
$45 - \left(\frac{L-2000}{50000-2000}\right)^{0.3} \times 40$	$\leq 50\,000$
0	$> 50\,000$

Ако направите невалидно действие или не познаете правилно снимките ще получите оценка 0 за теста. Иначе, оценката е равна на  $P_C \times P_L$ . Например на тест при  $T = 2$ , ако  $L = 2000$ , а  $P = 3$ , оценката Ви ще бъде  $45\% \times 60\% = 0.27$ .

### Ограничения

- $15 \leq N \leq 50\,000$
- $1 \leq T \leq 2$

### Подзадачи

Подзадача	Точки	Необходими подзадачи	$T$	Други ограничения
1	40	—	$= 1$	—
2	10	1	$= 2$	Номерата на двете снимки са последователни числа.
3	50	1 – 2	$\leq 2$	—

Точките за дадена подзадача се получават само ако се преминат успешно всички тестове, предвидени за нея и необходимите подзадачи, и са равни на минималната оценка на тест в нея, умножена по точките на подзадачата.

### Примерна комуникация

Функция на участника	Програма на журито
	play(6, 1)
guess({ {1,2,3}, {4,5,6} })	10
guess({ {1}, {2}, {3} })	010
return {2}	Correct!

Функция на участника	Програма на журито
	play(6, 2)
guess({ {1,2}, {3,4}, {5,6} })	011
guess({ {3}, {5} })	01
return {5,4}	Correct!

### Пояснение

Нека разгледаме първия пример. След първите два въпроса, зададени наведнъж, знаем че търсената снимка е измежду първите 3. Съответно с последните 3 въпроса откриваме и коя точно е.

### Локално тестване

За локално тестване е предоставен файлът `Lgrader.cpp`. Сложете Вашия файл `twins.cpp` заедно с `Lgrader.cpp` и `twins.h` в една папка и компилирайте трите файла заедно. Така ще получите програма, с която да проверите верността на функцията Ви. Програмата ще изисква две числа –  $N$  и  $T$ . На следващия ред се въвеждат номерата на снимките, които са на Рени. Програмата ще извежда подробно стъпките при изпълнение на зададения тест. За да промените това, може да смените стойността на константата `DETAILS` в началото на кода. Може да модифицирате този файл, както пожелаете.