



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Баня, 22 юни 2024 г.

Група А

Задача АТ11. МАГИЯ

1 сек. 256 MB

Магьосникът Хари е измислил нов интригуващ фокус с помощта на две превъзходни асистентки. Започва се с тесте с $2N + 1$ карти, номерирани с числата от 0 до $2N$. След което се извиква някой зрител от публиката, който първо трябва да избере една от картите и да я махне от тестето. После той разделя останалите $2N$ карти на две части с по N карти, като първата част раздава на първата асистентка, а втората част на втората асистентка. След това, първата асистентка показва две от дадените ѝ карти, без да има контакт с Хари или с другата асистентка (също може да предполагате, че картите се показват една по една, т.е. говорим за наредена двойка карти). Аналогично втората асистентка също показва две от дадените ѝ карти на магьосника. На база тази информация и малко магия, Хари винаги успява да познае коя е избраната карта от зрителя.

Задача

Вие искате да докажете, че магия не съществува и затова ще напишете програма **magic**, която да извършва описания фокус като изпълнява ролята на асистентките, както и на магьосника Хари. Единствено знаете, че предварително асистентките и магьосника Хари са изкомуникирали своята стратегия, но след като започне фокусът няма комуникация между тях.

Детайли по имплементацията

Трябва да имплементирате три функции (съответно за първата асистентка, за втората асистентка и за магьосника), които са в следния формат:

```
pair <int, int> assistant1 (vector <int>& cards);  
pair <int, int> assistant2 (vector <int>& cards);  
int magician (int N, pair <int, int> chosen1, pair <int, int>  
chosen2);
```

Програмата ви ще обработи T тестови случая, като в рамките на един тестов случай първо ще бъде извикана функцията `assistant1`, след това функцията `assistant2` и накрая функцията `magician`. Освен това в рамките на един тестов случай всяко извикване на функция ще е в различно изпълнение на вашата програма, за да се гарантира, че трите функции няма да споделят данни помежду си.

Функцията `assistant1` ще бъде извикана веднъж от програмата на журито (в рамките на един тестов случай) с един параметър - вектор, в който са записани раздадените N карти от зрителя на първата асистентка. Функцията трябва да върне наредена двойка с номера на две от дадените карти (номерата трябва да са различни), които ще бъдат показани на магьосника. Аналогично функцията `assistant2` трябва да играе роля на втората асистентка, която получава като параметър - вектор с раздадените N карти от зрителя и трябва да върне наредена двойка с номерата на избраните две различни карти от дадените, които да бъдат показани.

Функцията `magician` ще бъде извикана веднъж от програмата на журито (в рамките на един тестов случай) с три параметъра - съответно стойността на N , една наредена двойка, която задава избраните и показани карти от първата асистентка и втора наредена двойка, която задава картите на втората асистентка. Функцията трябва да върне едно число, което задава номера на избраната карта от зрителя.

Вашата програма **magic** трябва да имплементира трите описани функции. Тя може да съдържа и друг код, функции и глобални променливи, необходими за работата Ви, но не трябва



ТРЕНИРОВЪЧНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Баня, 22 юни 2024 г.

Група А

да съдържа главната функция `main`. Също така, не трябва да четете от стандартния вход или да отпечатвате на стандартния изход.

Ограничения

- ♣ $1 \leq T \leq 60\,000$;
- ♣ $6 \leq N \leq 1\,300\,000$;
- ♣ $\Sigma N \leq 1\,300\,000$.

Подзадачи

Подзадача	Точки	N	T	ΣN	Други ограничения
1	29	$= 6$	$\leq 15\,000$	$\leq 130\,000$	–
2	19	≤ 30	$\leq 15\,000$	$\leq 130\,000$	–
3	30	≤ 500	$\leq 15\,000$	$\leq 130\,000$	За всеки тест, най-много в 10 от тестовите случаи $N > 50$.
4	22	$\leq 1\,300\,000$	$\leq 60\,000$	$\leq 1\,300\,000$	–

Точките за дадена подзадача се получават само ако се преминат **успешно** всички тестове, предвидени за нея. Един тест е успешен, ако за всеки тестов случай сте спазили протокола за комуникация и накрая сте върнали правилно избраната карта от зрителя.

Примерна комуникация

В примерната комуникация има само един тестов случай, т.е. $T = 1$ и $N = 6$.

Действия на вашата програма	Действия и отговори на журито
	<code>assistant1({6, 1, 2, 5, 7, 10})</code>
<code>return {1, 2}</code>	
	<code>assistant2({3, 0, 4, 9, 12, 8})</code>
<code>return {4, 3}</code>	
	<code>magician(6, {1, 2}, {4, 3})</code>
<code>return 11</code>	

Локално тестване

За локално тестване са предоставени файловете `Lgrader.cpp` и примерен файл `magic.cpp`, в който може да напишете вашата програма. Когато предоставените файлове са в една папка, може да компилирате заедно вашата програма `magic.cpp` и `Lgrader.cpp`. Така ще получите програма, с която можете да проверите верността на функцията `Vi`.

Програмата ще изисква от стандартния вход следната последователност от данни:

- на първия ред: едно цяло положително число – броя тестови случаи T ;
- за всеки тестов случай на първия ред: едно цяло положително число - N ; а на втория/третия ред: по N различни положителни числа от 0 до $2N$, които задават раздадените карти на първата/втората асистентка.

Ако даден тестов случай не е успешен, то ще получите подходящо съобщение за грешка. Иначе, при успешен тестов случай, ще получите съобщение "Correctly guessed card k .", където k е номерът на избраната карта от зрителя.