

Санкт-Петербургский национальный исследовательский университет ИТМО

Факультет систем управления и робототехники

Алгоритмы и структуры данных

Отчёт по лабораторной работе №12 (1494)

Преподаватель: Тропченко А. А.

Выполнил: Марухленко Д. С.

Группа: R3235

Санкт Петербург, 2021г.

1. Цель работы

Решить задачу №1494 на платформе Timus Online Judge

<https://acm.timus.ru/problem.aspx?space=1&num=1494>

2. Задача

Условие

Стол для монобильярда, установленный в игровом доме уездного города N, оказался очень прибыльным вложением. До того, как в городе появился небезызвестный господин Чичиков. Раз за разом он выигрывал, и хозяин, подсчитывая убытки, понимал, что дело тут нечисто. Однако уличить подлеца в жульничестве не удавалось до прибытия в город N ревизора из Петербурга.

Правила игры в монобильярд очень просты: нужно последовательно закатить в единственную лузу шары с номерами 1, 2, ..., N (именно в этом порядке). Пока господин Чичиков играл, ревизор несколько раз подходил к столу и забирал из лузы последний закатившийся туда шар. В конце концов, оказалось, что Чичиков закатил в лузу все шары, а ревизор все шары достал и обследовал. Аферист утверждал, что закатил шары в правильном порядке. Хозяин понял, что это его шанс: ревизор должен помнить, в каком порядке он доставал шары. Однако так ли легко будет доказать жульничество?

Ограничение времени: 1.0 секунды

Ограничение памяти: 64 МБ

Исходные данные

В первой строке записано целое число N — количество бильярдных шаров ($1 \leq N \leq 100000$). В следующих N строках даны номера этих шаров в том порядке, в котором ревизор забирал их из лузы.

Результат

Выведите слово «Cheater», если Чичиков не мог закатить все N шаров в правильном порядке. Иначе выведите «Not a proof».

Пример

Исходные данные	Результат
2 2 1	Not a proof
3 3 1 2	Cheater

3. Материалы работы

3.1. Объяснение алгоритма

Для решения задачи создадим массив структур Interval, в которых будут храниться взятые мячи. По ходу взятия шаров проверяется время (порядковый номер), когда он был вынут. При этом в структуру записывается вынутый шар и допустимый для вытаскивания следующий шар. При доставании следующего шара проверяется его номер и «время», когда он был высунут и проверен. Если новый шар попадает в допустимый интервал, то проверка продолжается, иначе – выводится сообщение и мошенничество.

3.2. Код программы.

```
1. #include <iostream>
2. using namespace std;
3.
4. struct Interval {
5.     unsigned int left, right;
6. };
7.
8. Interval ballss[50000];
9. Interval * balls = ballss;
10. int main() {
11.     int n;
12.     cin >> n;
13.     unsigned int time = 0, take_ball;
14.     while (--n >= 0) {
15.         cin >> take_ball;
16.         if (take_ball > time) {
17.             if (take_ball > time + 1) {
18.                 (++balls)->left = time + 1;
19.                 balls->right = take_ball - 1;
20.                 time = take_ball;
21.             } else
22.                 ++time;
23.         } else {
24.             if (take_ball != balls->right) {
25.                 cout << "Cheater";
26.                 return 0;
27.             }
28.             if (--balls->right < balls->left) --balls;
29.         }
30.     }
31.     cout << "Not a proof";
32.     return 0;
33. }
```

4. Результат выполнения и ссылка на репозиторий GitHub

9259594	16:25:50 9 мар 2021	Daniil Marukhlenko	1494	G++ 9.2 x64	Accepted	0.125	776 КБ
-------------------------	------------------------	------------------------------------	----------------------	-------------------	----------	-------	--------

https://github.com/japersik/algorithms_and_data_structures/



5. Вывод

Работа выполнена, задача решена с использованием созданной структуры Interval.