

Отчет

по лабораторной работе «1494. Монобильярд»
по дисциплине «Алгоритмы и Структуры данных»

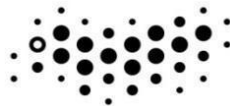
Авторы:

Полит Алексей Денисович

Факультет: СУиР

Группа: R3235

Преподаватель: Тропченко Андрей Александрович



УНИВЕРСИТЕТ ИТМО

1. Задачи

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?

2. 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.

3. Текст программы

```
#include <iostream>
#include <stack>
using namespace std;

int main() {
    int n, max = 0;
    cin >> n;
    stack<int> balls;
    for (int i = 0; i < n; i++) {
        int currentBall;
        cin >> currentBall;
        if (currentBall > max) {
            for (int j = max + 1; j <= currentBall - 1; j++) balls.push(j);
            max = currentBall;
        } else {
            if (currentBall == balls.top()) balls.pop();
            else {
                cout << "Cheater" << endl;
            }
        }
    }
}
```

```

        return 0;
    }
}
}
cout << "Not a proof" << endl;
return 0;
}

```

9276976	13:27:55 22 мар 2021	Aleksey	1494. Монобильярд	G++ 9.2 x64	Accepted	0.156	832 КБ
---------	-------------------------	-------------------------	-----------------------------------	-------------	----------	-------	--------

4. Описание программы

Основная сложность задачи - понять крайне запутанное условие.

Возьмём множество шаров $x[1], x[2], x[3] \dots x[n]$.

Игроку нужно забивать шары строго в порядке возрастания номеров.

Значит, ревизору нужно доставать их строго

в обратном порядке, чтобы подтвердить честность игрока. При этом, ревизор может подойти в любой момент времени.

Первый раз он подходит и достаёт шар $x[k]$ и предполагает, что Чичиков забил все $\{x[i < k]\}$ шары.

Когда он подойдёт следующий раз и вытащит шар $x[m]$, есть 3 варианта развития событий:

1. Если $(m - k > 1)$, то ревизор может предположить, что Чичиков забил все шары на промежутке между $x[k]$ и $x[m]$,

и это не будет являться доказательством виновности.

2. Если $(m - k = 1)$, значит за время отсутствия ревизора, Чичиков ничего не забил, а ревизор просто достал предыдущий шар.

Обвинить Чичикова нельзя.

3. Если $((m - k) \leq 0)$, значит ревизор достал шар, на несколько номеров меньше, чем предыдущий взятый. Значит, за время отсутствия ревизора, Чичиков забил шар с меньшим номером, или не забивал вообще, а нарушил порядок раньше.

Это будет являться доказательством его виновности.

3 4 2 1 5 6

Достаём шар 3 -> предполагаем, что Чичиков забил 1..3

Достаём шар 4 -> Чичиков забил 4

Достаём шар 2 -> Чичиков ничего не забил за время отсутствия ревизора

Достаём шар 1 -> Чичиков ничего не забил за время отсутствия ревизора

Достаём шар 5 -> Чичиков забил 5

Достаём шар 6 -> Чичиков забил 6

НЕТ ДОКАЗАТЕЛЬСТВ ВИНЫ

3 1 2

Достаём шар 3 -> предполагаем, что Чичиков забил 1..3

Достаём шар 1 -> если Чичиков забивал 3, значит по правилам, до него должен был забит 2, а тут 1

ЧИЧИКОВ ЖУЛЬНИЧАЕТ

Реализовать такую проверку ревизором можно через структуру данных - стек. Достанем $x[k]$ и положим в стек, все шары, которые должен был забить Чичиков $\{x[i < k]\}$, если $x[k]$ является $x[\max]$. Иначе проверим, является ли взятый шар ожидаемым, т.е. для шара $x[k]$, шар с вершины стека должен быть $x[k-1]$. Если условие выполняется, то всё в порядке, иначе вынесем обвинительный приговор.