

ВШЭ АиСД 2021. STL!

12 янв 2022, 18:09:50

старт: 7 дек 2021, 20:00:00

финиш: 18 дек 2021, 03:59:00

длительность: 10д. 7ч.

начало: 7 дек 2021, 20:00:00

конец: 18 дек 2021, 03:59:00

# G. Возрастающая подпоследовательность (0.25)

Ограничение времени	1 секунда
Ограничение памяти	64.0 Мб
Ввод	стандартный ввод
Вывод	стандартный вывод

Дан массив целых чисел, каждое из которых по модулю не превосходит  $10^9$ .

Требуется определить размер самой длинной последовательности, которую можно образовать из исходного массива так, что числа в этой последовательности следуют строго друг за другом (каждое следующее больше предыдущего на 1).

## Формат ввода

В первой строке задается число  $n$  ( $1 \leq n \leq 8 \cdot 10^5$ ) - размер массива.

В следующей строке  $n$  чисел, каждое из которых по модулю не превосходит  $10^9$ .

## Формат вывода

Выведите одно целое число – размер самой длинной последовательности.

### Пример 1

Ввод	Вывод
6 1 2 3 6 5 4	6

### Пример 2

Ввод	Вывод
6 4 2 3 8 6 5	5

### Пример 3

Ввод	Вывод
6 -2 -3 -1 7 6 0	4

Примечания

Система оценки

Группа	Баллы	Доп. ограничения	Необх. группы	Комментарий
		$n$		
0	1	–	–	Тесты из условия.
1	2	$n \leq 10^4$	0	
2	2	$n \leq 10^5$	0 – 1	
3	3	$n \leq 5 \cdot 10^5$	0 – 2	
4	2	$n \leq 8 \cdot 10^5$	0 – 3	Offline-проверка

Набрать здесь

Отправить файл

```
1 #include <iostream>
2 #include <algorithm>
3 #include <set>
4
5 void findMaxSequence(std::set<int> *set) {
6     int maximum_sequence = 1;
7     int maximum_actual = 1;
8
9     auto pre_end = prev(set->end());
10    for (auto iterator = set->begin(); iterator != pre_end; ++iterator) {
11        if (*iterator + 1 == *next(iterator)) {
12            ++maximum_actual;
13        } else {
14            maximum_actual = 1;
15        }
16
17        maximum_sequence = std::max(maximum_sequence, maximum_actual);
18    }
19    std::cout << maximum_sequence;
20 }
21
22 int main() {
23     std::ios_base::sync_with_stdio(false);
24     std::cin.tie(nullptr);
25
26     int number_of_elements;
27     std::cin >> number_of_elements;
28     std::set<int> set;
29
30     for (int i = 0; i < number_of_elements; ++i) {
31         int number;
32         std::cin >> number;
33         set.insert(number);
34     }
35
36     findMaxSequence(&set);
37     return 0;
38 }
```

Отправить

Предыдущая