

# 15. Великое Лайнландское переселение

Ограничение времени	1 секунда
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Лайнландия представляет из себя одномерный мир, являющийся прямой, на котором располагаются  $N$  городов, последовательно пронумерованных от  $0$  до  $N - 1$ . Направление в сторону от первого города к нулевому названо западным, а в обратную — восточным. Когда в Лайнландии неожиданно начался кризис, все были жители мира стали испытывать глубокое смятение. По всей Лайнландии стали ходить слухи, что на востоке живётся лучше, чем на западе.

Так и началось Великое Лайнландское переселение. Обитатели мира целыми городами отправились на восток, покинув родные улицы, и двигались до тех пор, пока не приходили в город, в котором средняя цена проживания была меньше, чем в родном.

## Формат ввода

В первой строке дано одно число  $N$  ( $2 \leq N \leq 10^5$ ) — количество городов в Лайнландии. Во второй строке дано  $N$  чисел  $a_i$  ( $0 \leq a_i \leq 10^9$ ) — средняя цена проживания в городах с нулевого по  $(N - 1)$ -ый соответственно.

## Формат вывода

Для каждого города в порядке с нулевого по  $(N - 1)$ -ый выведите номер города, в который переселятся его изначальные жители. Если жители города не остановятся в каком-либо другом городе, отправившись в Восточное Бесконечное Ничто, выведите  $-1$ .

## Пример

<b>Ввод</b> <input type="text"/>	<b>Вывод</b> <input type="text"/>
10 1 2 3 2 1 4 2 5 3 1	-1 4 3 4 -1 6 9 8 9 -1

Язык 

GNU GCC 12.2 C++20

Набрать здесь

Отправить файл

```
1 #include <iterator>
2 #include <utility>
3 #include <iostream>
4 #include <vector>
5 #include <stack>
6
7 int main() {
8     int n;
9     std::cin >> n;
10    std::vector<int> ret;
11    ret.resize(n);
12    std::stack<std::pair<int, int>> stk;
13
14    for (int i = 0, value; i != n; ++i) {
15        std::cin >> value;
16        while (!stk.empty()) {
17            auto curr = stk.top();
18            if (value < curr.first) {
19                ret[curr.second] = i;
20                stk.pop();
21            } else {
22                break;
23            }
24        }
25        stk.push({value, i});
26    }
27    while (!stk.empty()) {
28        auto curr = stk.top();
29        ret[curr.second] = -1;
30        stk.pop();
31    }
32
33    std::copy(ret.begin(), ret.end(), std::ostream_iterator<int> (std::cout, " "));
34    return 0;
35 }
```

Отправить

Предыдущая

Следующая

---

© 2013–2023 ООО «Яндекс»