

Лабораторная работа 5

Массивы - 1

Цель этой лабораторной работы — научиться пользоваться STL классом `vector` в качестве массива для эффективной работы с данными в C++.

(!) Обычными (тупыми) массивами пользоваться нельзя!

В задачах, где входными данными является последовательность чисел, последовательность даётся в формате $n \ a_0 \ a_1 \ \dots \ a_{n-1}$. Т.е. первое число — это количество элементов последовательности (само число не входит).

Вы **должны создать** `vector` из данной последовательности при вводе. Далее этот вектор должен передаваться как аргумент в функцию `solve`. Функция `solve` unit-тестируется вами.

В задачах, где требуется поменять массив — вы должны именно **поменять** сам массив, и затем вывести его!

Задания А

1. Дан массив целых чисел. Посчитайте, сколько в массиве элементов, больших предыдущего. Выведите ответ.

Напишите функцию

Пример. Ввод: 7 -1 2 -3 4 -5 -6 7 Вывод: 3

2. Дан массив целых чисел. Если в нем есть два соседних элемента одного знака, выведите эти числа. Если соседних элементов одного знака нет — не выводите ничего. Если таких пар соседей несколько — выведите первую пару.

Пример. Ввод: 7 -1 2 -3 4 -5 -6 7 Вывод: -5 -6

3. Дан массив целых чисел. В заданном одномерном массиве поменять местами соседние элементы, стоящие на четных местах, с элементами, стоящими на нечетных. Вывести полученный массив.

Пример. Ввод: 7 4 76 -1 -7 15 -33 5 Вывод: 76 4 -7 -1 -33 15 5

4. Дан массив a из n целых чисел. Дан массив b из n целых чисел. Поменяйте местами соответствующие элементы a_i, b_i так, что всегда будет выполняться условие $a_i \leq b_i$ для каждого i . Выведите получившиеся массивы.

5. Дана последовательность натуральных чисел от 1 до 9. Посчитайте в этой последовательности количество единиц, количество двоек, количество троек и т.д. и выведите результат. В выходных данных всегда должно быть 9 чисел.

Пример. Ввод: 11 1 1 4 1 5 8 6 3 5 1 6 Вывод: 4 0 1 1 2 2 0 1 0

Задания В

1. Дан массив натуральных чисел. Есть ли среди них простые числа? Если да, то вывести индексы этих элементов, в одной строке через пробел. Если нет, ничего не выводить. Напишите и используйте функцию `is_prime` для проверки чисел на простоту.

Пример. Ввод: 1 2 3 4 5 6 7 8 9 10 Вывод: 1 2 3 6

2. Дан массив целых чисел. Удалить из массива те числа, которые удалены от наибольшего **и** наименьшего чисел массива на расстояние, большее 1. Вывести полученный массив. Решите эту задачу через **создание нового массива** и замены им оригинального массива.

Пример. Ввод: 6 1 3 5 4 2 6 Вывод: 1 5 2 6

3. Дан массив x из n вещественных чисел. Дан массив y из n вещественных чисел. Найдите и выведите *наименьшее* расстояние между двумя точками этого множества точек (x_i, y_i) . Для решения задачи, напишите вспомогательную функцию `dist(x_a, y_a, x_b, y_b)`, которая будет вычислять расстояние между точками (x_a, y_a) и (x_b, y_b) на плоскости.

4. Дан массив целых чисел. Не используя дополнительные массивы, определить, какой элемент встречается чаще всех. Гарантируется, что есть элемент, который строго чаще всех других встречается.

Пример. Ввод: 9 1 2 3 2 3 3 2 4 3 Вывод: 3

5. Дан массив. Выведите только те значения элементов массива, которые встречаются в нём ровно один раз. Элементы следует распечатывать в том порядке, в котором они встречаются в массиве.

Пример. Ввод: 8 4 8 16 4 2 -8 -8 4 Вывод: 8 16 2