

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра Название кафедры

ОТЧЕТ
по лабораторной работе №2
по дисциплине «Основы алгоритмизации и программирования на языке
С++»
Тема: ОДНОМЕРНЫЕ СТАТИЧЕСКИЕ МАССИВЫ

Студент гр. 1324

Скопцов В.В.

Преподаватель

Глущенко А.Г.

Санкт-Петербург

2021

Цель работы.

Необходимо написать программу, которая:

- 1) Создает целочисленный массив размерности $N = 100$. Элементы массивы должны принимать случайное значение в диапазоне от -99 до 99.
- 2) Отсортировать заданный в пункте 1 массив [...] сортировкой (от меньшего к большему). Определить время, затраченное на сортировку, используя библиотеку `chrono`.
- 3) Найти максимальный и минимальный элемент массива. Подсчитайте время поиска этих элементов в отсортированном массиве и неотсортированном, используя библиотеку `chrono`.
- 4) Выводит среднее значение (если необходимо, число нужно округлить) максимального и минимального значения. Выводит индексы всех элементов, которые равны этому значению, и их количество.
- 5) Выводит количество элементов в отсортированном массиве, которые меньше числа a , которое инициализируется пользователем.
- 6) Выводит количество элементов в отсортированном массиве, которые больше числа b , которое инициализируется пользователем.
- 7) Выводит информацию о том, есть ли введенное пользователем число в отсортированном массиве. Реализуйте алгоритм бинарного поиска. Сравните скорость его работы с обычным перебором. (*)
- 8) Меняет местами элементы массива, индексы которых вводит пользователь. Выведите скорость обмена, используя библиотеку `chrono`.

Основные теоретические положения.

При использовании простых переменных каждой области памяти для хранения данных соответствует свое имя. Если с группой величин одинакового типа требуется выполнить однообразные действия, им дают одно имя, а различают по порядковому номеру (индексу). Это дает возможность компактно записать множество операций с использованием циклов.

Массив представляет собой индексированную последовательность однотипных элементов с заранее определенным количеством элементов. Наглядно одномерный массив можно представить как набор пронумерованных ячеек, в каждой из которых содержится определенное значение.

Все массивы можно разделить на две группы: одномерные и многомерные. Описание массива в программе отличается от объявления обычной переменной наличием размерности массива, которая задается в квадратных скобках после имени.

Элементы массива нумеруются с нуля. При описании массива используются те же модификаторы (класс памяти, const и инициализатор), что и для простых переменных.

Результаты выполнения.

Примеры выполнения программ:

```
hav@hav-N53SV:~/Музыка/Praktika/Praktika$ ^C
hav@hav-N53SV:~/Музыка/Praktika/Praktika$ g++ laboratornaya2.cpp
hav@hav-N53SV:~/Музыка/Praktika/Praktika$ ./a.out
Paragraph №1
-56 78 -12 -28 -79 -84 -70 -77 30 65 -97 26 -77 -41 -43 85 -84 59 46 20 -65 72 -85 -13 76 -87 26
76 78 75 90 99 54 -45 -52 51 -30 -46 73 77 -80 53 -20 -58 -88 13 -96 -96 -50 50 1 -16 -1 15 70 -2
5 28 -26 -72 -17 -51 94 -41 79 -51 6 8 -5 60 -19 -28 56 -66 28 74 -78 -58 77 -74 -9 4 26 74 79 -8
2 21 -46 22 71 -42 82 97 -48 -59 76 -23 24 -16 72 -39
Again? 1 - yes, 0 - no 0
```

Рисунок 1

```
ТЕРМИНАЛ BASH
Paragraph №2
-97 -96 -96 -88 -87 -85 -84 -84 -82 -80 -79 -78 -77 -77 -74 -72 -70 -66 -65 -59 -58 -58 -56 -52 -
51 -51 -50 -48 -46 -46 -45 -43 -42 -41 -41 -39 -30 -28 -28 -26 -25 -23 -20 -19 -17 -16 -16 -13 -1
2 -9 -5 -1 1 4 6 8 13 15 20 21 22 24 26 26 26 28 28 30 46 50 51 53 54 56 59 60 65 70 71 72 72 73
74 74 75 76 76 77 77 78 78 79 79 82 85 90 94 97 99
Again? 1 - yes, 0 - no 0
```

Рисунок 2

```
Paragraph №3
Max 99, Min -97
Duration of the Search in unsorted array 0.00213ms.
Duration of the Search in sorted array 0.000186ms.

Again? 1 - yes, 0 - no 0
```

Рисунок 3

```
Paragraph №4
Average value 1
1[52]
The number of elements equal to the average 1

Again? 1 - yes, 0 - no 0
```

Рисунок 4

```
Paragraph №5
Enter the number 35
The number of elements less than specified value 68

Again? 1 - yes, 0 - no
0
```

Рисунок 5

```
Paragraph №6
Enter the number 35
The number of elements larger than specified value 32

Again? 1 - yes, 0 - no 0
```

Рисунок 6

```
Paragraph №7
Enter the number 35
Not found
Duration of the Binary search 0.000967ms.
Duration of the Brute-force search 0.001579ms.

Again? 1 - yes, 0 - no 0
```

Рисунок 7

```
Paragraph №8
Enter the indexes 0 99
99 -96 -96 -88 -87 -85 -84 -84 -82 -80 -79 -78 -77 -77 -74 -72 -70 -66 -65 -59 -58 -58 -56 -52 -5
1 -51 -50 -48 -46 -46 -45 -43 -42 -41 -41 -39 -30 -28 -28 -26 -25 -23 -20 -19 -17 -16 -16 -13 -12
-9 -5 -1 1 4 6 8 13 15 20 21 22 24 26 26 26 28 28 30 46 50 51 53 54 56 59 60 65 70 71 72 72 73 7
4 74 75 76 76 76 77 77 78 78 79 79 82 85 90 94 97 -97
Duration of the Exchange 0.000942ms.
```

Рисунок 8

Выводы.

В результате выполнения работы была написанна программа, которая:

- 1) Создает целочисленный массив размерности $N = 100$. Элементы массивы должны принимать случайное значение в диапазоне от -99 до 99.
- 2) Отсортировать заданный в пункте 1 массив [...] сортировкой (от меньшего к большему). Определить время, затраченное на сортировку, используя библиотеку chrono.
- 3) Найти максимальный и минимальный элемент массива. Подсчитайте время поиска этих элементов в отсортированном массиве и неотсортированном, используя библиотеку chrono.
- 4) Выводит среднее значение (если необходимо, число нужно округлить) максимального и минимального значения. Выводит индексы всех элементов, которые равны этому значению, и их количество.
- 5) Выводит количество элементов в отсортированном массиве, которые меньше числа a , которое инициализируется пользователем.
- 6) Выводит количество элементов в отсортированном массиве, которые больше числа b , которое инициализируется пользователем.
- 7) Выводит информацию о том, есть ли введенное пользователем число в отсортированном массиве. Реализуйте алгоритм бинарного поиска. Сравните скорость его работы с обычным перебором. (*)
- 8) Меняет местами элементы массива, индексы которых вводит пользователь. Выведите скорость обмена, используя библиотеку chrono.

Таким образом, программа полностью соответствует условиям поставленной задачи и цели лабораторной работы.