

**Л. Н. ТРОФИМОВА, А. В. ЕРОШЕНКО**

**ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++**

***ЧАСТЬ 3***

**ОМСК 2018**

Министерство транспорта Российской Федерации  
Федеральное агентство железнодорожного транспорта  
Омский государственный университет путей сообщения

---

Л. Н. Трофимова, А. В. Ерошенко

## ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++

### *Часть 3*

Утверждено методическим советом университета  
в качестве учебно-методического пособия  
к выполнению лабораторных работ и самостоятельной работы  
по дисциплине «Информатика»

Омск 2018

УДК 004. 432 (075.8)  
ББК 22.183.492я73  
Т76

**Основы программирования на языке C++:** Учебно-методическое пособие к выполнению лабораторных работ и самостоятельной работы по дисциплине «Информатика». Часть 3/ Л. Н. Трофимова, А. В. Ерошенко; Омский гос. ун-т путей сообщения. Омск, 2018. 28 с.

Содержит основы программирования на языке C++. Рассматриваются различные способы ввода и вывода одномерных числовых массивов. Приводятся практические рекомендации для решения задач обработки числовых массивов. Представлены задания для аудиторной и самостоятельной работы студентов по написанию программ выполнения стандартных действий над одномерными массивами.

Предназначено для студентов первого курса Института автоматизи-рованных коммуникаций и информационных технологий всех специальностей очной и заочной форм обучения.

Библиогр.: 4 назв. Табл. 5. Рис. 1.

Рецензенты: канд. пед. наук, доцент Л. А. Усольцева;  
канд. техн. наук, доцент А. Г. Малютин.

## ОГЛАВЛЕНИЕ

Введение .....	5
Лабораторная работа 9. Одномерные массивы.....	6
9.1. Понятие массива. Объявление, ввод (вывод) элементов одномерного массива .....	6
9.2. Стандартные действия над одномерными числовыми массивами.....	11
9.3. Задания.....	18
Библиографический список.....	27



## ВВЕДЕНИЕ

Учебно-методическое пособие посвящено важнейшему разделу информатики – программированию на ЭВМ – и является логическим продолжением первой и второй частей учебно-методических пособий [3, 4].

Данное учебно-методическое пособие составлено в соответствии с рабочей программой дисциплины «Информатика», предназначено для студентов первого курса, изучающих программирование, и содержит материал к выполнению лабораторных работ по дисциплине «Информатика», а также для самостоятельного изучения языка программирования C++.

Цель данной работы – ознакомить студентов с основами программирования числовых массивов на языке C++, рассмотреть основные способы обработки одномерных массивов и научить творческому подходу к решению поставленных задач.

Пособие включает в себя теоретические сведения, примеры стандартных действий над одномерными числовыми массивами и пояснения к ним, задания для аудиторной и самостоятельной работы.

## ОДНОМЕРНЫЕ МАССИВЫ

Цель работы: изучение особенностей работы с одномерными числовыми массивами.

### 9.1. Понятие массива. Объявление, ввод (вывод) элементов одномерного массива

*Массив* – это совокупность данных, обозначаемых одним именем (идентификатором) и хранящихся вместе. Отдельные данные в массиве называются *элементами массива*. Номера элементов массива начинаются с нуля. Каждому элементу массива ставятся в соответствие *индексы*.

Количество индексов указывает на *размерность массива*. Массивы по размерности бывают одномерными, двумерными, многомерными.

Объявляя массив, необходимо сначала указать тип хранимых данных, а затем имя массива и его размер.

Общая форма описания *N*-мерного массива имеет вид:

*Тип массива имя массива* [размер 1] [размер 2] ... [размер *N*];

Одномерный массив содержит в описании только один индекс:

*Тип массива имя массива* [размер 1];

Количество фактических элементов в массиве не должно превышать количества объявленных элементов. К каждому из элементов массива можно обратиться по его номеру (индексу), расположенному в квадратных скобках после имени массива.

Существует несколько способов ввода и вывода элементов массива, которые рассмотрены в примерах ниже.

**Пример 1.** Объявить массив целого типа. Значения элементов массива инициализировать в программе. Вывести элементы массива с указанием индексов в строку. При выводе применить циклическую конструкцию.

```
#include "stdafx.h"
#include<iostream>
const int N = 10; // именованная константа N
using namespace std;
```

```

int _tmain(int argc, _TCHAR* argv[])
{
    int mas[N] = { 11, 12, 4, 5, 3, 9, 55, 1, 0, 7 }; /* int – целый тип, mas –
имя массива, N – количество элементов*/
    int i; // i – индекс массива
    for (i = 0; i < N; i++)
    {
        cout << "mas[" << i << "]=" << mas[i] << " "; /*вывод эле-
ментов массива в строку*/
    }
    return 0;
}

```

Пр и м е р 2. Объявить массив вещественного типа. Значения элементов массива инициализировать в программе. Вывести элементы массива в столбец.

```

#include "stdafx.h"
#include<iostream>
const int N = 10;
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
    double mas[N] = { 11.4, 12.5, 4.2, 5.6, 3., 9.7, 5.5, 1.9, 0.5, 7.1 };
/*double – вещественный тип двойной точности, mas – имя массива, N – коли-
чество элементов*/
    int i;
    for (i = 0; i < N; i++)
    {
        cout << "mas[" << i << "]=" << mas[i] << '\n'; /* вывод эле-
ментов массива в столбец*/
    }
    return 0;
}

```

Пр и м е р 3. Ввести элементы массива с клавиатуры с использованием циклического алгоритма. Вывести элементы массива с указанием индексов в столбец.



```

#include "stdafx.h"
#include<iostream>
const int N = 10;
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
    double mas[N];
    int i;
    for (i = 0; i < N; i++)
    {
        cout << "Vvedite element massiva" << '\n';
        cin >> mas[i];
    }
    for (i = 0; i < N; i++)
    {
        cout << "mas[" << i << "]=" << mas[i] << '\n';
    }
    return 0;
}

```

Графическая схема алгоритма примера 3 представлена на рисунке.

**Пример 4.** Ввести элементы массива целого типа с использованием генератора случайных величин. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести элементы массива в строку.

```

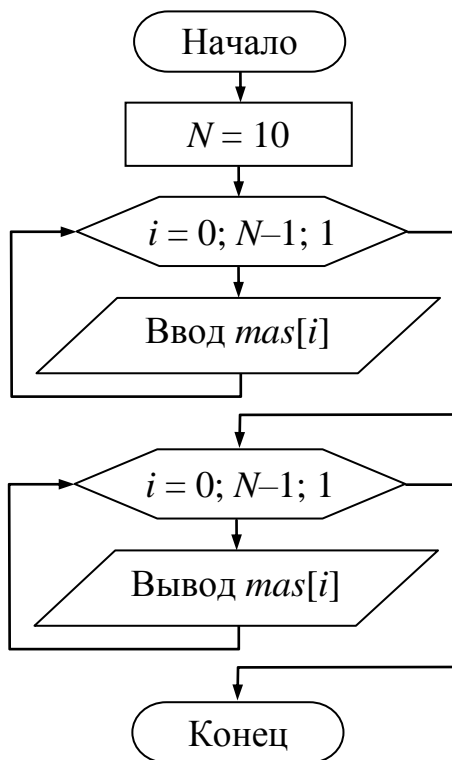
#include "stdafx.h"
#include<iostream>
#include<stdlib.h>> // подключение стандартной библиотеки
using namespace std;
const int M = 8;
int _tmain(int argc, _TCHAR* argv[])
{
    int mas[M], i, N = 24;
    for (i = 0; i < M; i++)
    {

```

```

        mas[i] = rand() % N; /*присвоение элементу массива случай-
ное целое число в диапазоне от 0 до N - 1*/
        cout << "mas[" << i << "]=" << mas[i] << " ";
    }
    return 0;
}

```



ГСА ввода и вывода  
элементов одномерного массива

Пример 5. Ввести элементы массива целого типа с использованием генератора случайных величин в диапазоне от 12 до 50. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести элементы массива в столбец.

```

#include "stdafx.h"
#include<iostream>
#include<stdlib.h>/*    подключение
стандартной библиотеки */
using namespace std;
const int M = 8;
int _tmain(int argc, _TCHAR* argv[])
{
    int mas[M], i;
    int a = 12; //начальное значение
                диапазона случайных чисел

```

```

    int b = 50; // конечное значение диапазона случайных чисел
    for (i = 0; i < M; i++)
    {
        mas[i] = rand() % (b - a + 1) + a; /*присвоение элементу мас-
сива случайного целого числа в диапазоне от 12 до 50*/
        cout << "mas[" << i << "]=" << mas[i] << '\n';
    }
    return 0;
}

```

Пример 6. Ввести элементы массива вещественного типа с использованием генератора случайных величин в диапазоне от  $-10$  до  $10$ . При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести элементы массива в столбец.

```
#include "stdafx.h"
#include<iostream>
#include<stdlib.h>
using namespace std;
const int N = 8;
int _tmain(int argc, _TCHAR* argv[])
{
    double mas[N];
    int i;
    for (i = 0; i < N; i++)
    {
        int n1 = rand() % 2001 - 1000; /* переменной целого типа n1 присваивается случайное целое число в диапазоне от  $-1000$  до  $1000$  */
        double x = n / 100.; /* вещественной переменной x присваивается частное от деления переменной n на 100. (после числа 100 необходимо ставить точку, чтобы показать, что деление производится на вещественное число). В результате переменная x будет принимать произвольное вещественное значение от  $-10$  до  $10$  */
        mas[i] = x; /*элементу массива присваивается случайное вещественное число в диапазоне от  $-10$  до  $10$ */
    }
    for (i = 0; i < N; i++)
    {
        cout << "mas[" << i << "]= " << mas[i] << '\n';
    }

    return 0;
}
```

Представленные способы ввода элементов массива с помощью генератора случайных величин при каждом запуске программы будут выдавать одни и те же числа, что удобно при отладке программы. Для того чтобы при каждом

запуске программы получать разный набор данных, применяют системные часы компьютера.

**Пример 7.** Ввести элементы массива целого типа с использованием генератора случайных величин. При каждом запуске программы элементы массива должны принимать другие значения. Вывести элементы массива в строку.

```
#include "stdafx.h"
#include <iostream>
#include <stdlib.h>
#include <time.h> // подключение библиотеки системных часов
using namespace std;
const int N = 12;
int _tmain(int argc, _TCHAR* argv[])
{
    int mas[N], i;
    time_t t; // инициализация генератора случайных чисел с помощью
    srand(time(&t)); // текущего календарного времени
    for (i = 0; i < N; i++)
    {
        mas[i] = rand() % 100;
        cout << "mas[" << i << "]=" << mas[i] << " ";
    }
    return 0;
}
```

## 9.2. Стандартные действия над одномерными числовыми массивами

Рассмотрим стандартные действия с одномерными числовыми массивами: поиск элемента массива, вставка и удаление элементов числового массива, формирование нового массива из элементов исходного массива по определенным правилам, определение минимального и максимального по значению элементов.

**Пример 8.** Ввести одномерный массив целого типа *mas[16]* с применением генератора случайных величин. Элементы массива должны находиться в диапазоне от -50 до 50. Проверить, является ли число, введенное пользователем с клавиатуры, элементом массива.

```

#include "stdafx.h"
#include<iostream>
#include<stdlib.h>
#include<time.h>
using namespace std;
const int n = 16;
int _tmain(int argc, _TCHAR* argv[])
{
    int i, mas[n];
    time_t t;
    srand(time(&t));
    for(i = 0; i < n; i++)
    {
        mas[i] = rand() % 101 - 50;
        cout<<mas[i]<<" ";
    }
    cout<<"\n";
    int r;
    cout<<"Vvedite chislo";
    cin>>r; // ввод произвольного числа
    for(i = 0; i < n; i++)
    {
        if(r == mas[i]) // если число равно i-му элементу массива
        {
            cout<<"YES"<<"\n"; // вывод «да»
            return 0; // конец программы
        }
    }
    cout<<"NO"<<"\n"; // иначе вывод «нет»
    return 0;
}

```

Пр и м е р 9. Одномерный массив целого типа *mas[6]* объявить в программе. Удалить элемент одномерного массива с индексом три.

Удаление одного элемента из одномерного массива происходит по следующей схеме:

определяется порядковый номер элемента, который необходимо удалить из массива;

все элементы, стоящие за удаляемым элементом, сдвигаются на одну позицию влево;

количество элементов уменьшается на единицу.

```
#include "stdafx.h"
#include<iostream>
using namespace std;
int _tmain(int argc, _TCHAR* argv[])
{
    double mas[6] = {0, 2, 3, 5, 6, 7};
    int i;
    for(i = 3; i < 5; i++)
    {
        mas[i] = mas[i + 1]; /* присвоение i-му элементу массива
значение i+1-го элемента */
    }
    for(i = 0; i < 5; i++)
    {
        cout<<mas[i]<< '\n';
    }
    return 0;
}
```

Пр и м е р 10. Между третьим и четвертым элементами одномерного числового массива вставить число 78.

При вставке одного элемента в массив необходимо сначала освободить место для вставляемого элемента, т. е. «раздвинуть» элементы, а затем вставить новый элемент в массив, при этом размер массива увеличивается на один элемент, что необходимо предусмотреть заранее.

```
#include "stdafx.h"
#include<iostream>
using namespace std;
```

```

int _tmain(int argc, _TCHAR* argv[])
{
    int i;
    double mas[7] = {1, 2, 3, 4, 5, 6};
    for(i = 5; i >= 3; i -= 1)
    {
        mas[i+1] = mas[i]; /* присвоение i+1-му элементу массива
значение i-го элемента */
    }
    mas[3] = 78;
    for(i = 0; i < 7; i++)
    {
        cout<<"mas["<<i<<"]="<<mas[i]<<"\t";
    }
    cout<<"\n";
    return 0;
}

```

Пр и м е р 11. Ввести одномерный массив целого типа *mas*[20] с применением генератора случайных величин. Элементы массива должны находиться в диапазоне от –50 до 50. Сформировать новый массив *mas1*, состоящий из положительных элементов исходного одномерного массива.

```

#include "stdafx.h"
#include<iostream>
#include<stdlib.h>
#include<time.h>
using namespace std;
const int n = 20;
int _tmain(int argc, _TCHAR* argv[])
{
    int i,mas[n],mas1[n];
    time_t t;
    srand(time(&t));
    for(i = 0; i < n; i++)
    {

```

```

        mas[i] = rand() % 101 - 50;
        cout<<mas[i]<< " ";
    }
    cout<<'\n';
    int j = 0;
    for(i = 0; i < n; i++)
    {
        if(mas[i] > 0) // если элемент массива положителен
        {
            mas1[j] = mas[i]; // формируем новый массив
            cout<<"mas1["<<j<<"]="<<mas1[j]<<" ";
            j++;
        }
    }
    return 0;
}

```

Пр и м е р 12. Задан одномерный массив целого типа *mas*[16]. Сформировать новый массив *mas2* из элементов исходного одномерного массива, которые имеют в младшем разряде целой части ноль.

Особенностью этой задачи является то, что размер нового формируемого массива неизвестен и не может быть заранее определен, поэтому для обозначения индексов исходного и нового массивов вводят две независимые переменные – *i* и *j*.

```

#include "stdafx.h"
#include<iostream>
#include<stdlib.h>
using namespace std;
const int n = 16;
int _tmain(int argc, _TCHAR* argv[])
{
    int mas[n], mas2[n], i;
    for(i = 0; i < n; i++)
    {
        mas[i] = rand() % 1001 - 500;
    }
}

```



```

        cout<<mas[i]<<" ";
    }
    cout<<"\n";
    int j = 0;
    for(i = 0; i < n; i++)
    {
        if(mas[i] % 10 == 0)
        {
            mas2[j] = mas[i]; // формируем новый массив
            cout<<mas2[j]<<" ";
            j++;
        }
    }
    return 0;
}

```

**Пример 13.** Найти сумму отрицательных элементов, расположенных между минимальным и максимальным элементами массива *mas*[16]. Найти произведение всех элементов массива, расположенных между минимальным и максимальным элементами массива. Отсортировать массив по возрастанию.

```

#include "stdafx.h"
#include<iostream>
#include<stdlib.h>
using namespace std;
const int n = 16;
int _tmain(int argc, _TCHAR* argv[])
{
    int i, mas[n];
    for(i = 0; i < n; i++)
    {
        mas[i] = rand() % 101 - 50;
        cout<<mas[i]<<" ";
    }
    cout<<"\n";
    int imax = 0, imin = 0;
    double max, min;

```

```

max = min = mas[0];
for(i = 1; i < n; i++)
{
    if(max < mas[i]) // если элемент массива больше max
    {
        max = mas[i]; // max присвоить элемент массива
        imax = i; // запомнить индекс этого элемента
    }
    if(min > mas[i]) // если элемент массива меньше min
    {
        min = mas[i]; // min присвоить элемент массива
        imin = i; // запомнить индекс этого элемента
    }
}
int delta;
double s = 0, p = 1;
if(imax > imin) /* если максимальный элемент расположен левее
минимального */
    delta = 1; // переменной delta присвоить 1
else delta = -1; // иначе переменной delta присвоить -1
for(i = imin; i != imax; i += delta) /* в промежутке между мини-
мальным и максимальным элементами */
{
    p = p * mas[i];
    if(mas[i] < 0)
        s = s + mas[i];
}
cout<<"s="<<s<<"\n"<<"p="<<p<<"\n";
for(i = 0; i < n - 1; i++)
{
    for(int j = i + 1; j < n; j++)
    {
        if(mas[i] > mas[j])
        {
            double x = mas[i];

```

```

        mas[i] = mas[j];
        mas[j] = x;
    }
}
for(i = 0; i < n; i++)
{
    cout<<"mas["<<i<<"]="<<mas[i]<<" ";
}
cout<<"\n";
return 0;
}

```

### 9.3. Задания

Задание 1. Разработать графическую схему алгоритма и программу ввода и вывода одномерных числовых массивов в соответствии с номером варианта (табл. 9.1).

Т а б л и ц а 9.1

Ввод и вывод одномерного числового массива

Номер варианта	Задание
1	2
1	Ввести одномерный массив, состоящий из 11 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в один столбец
2	Ввести одномерный массив, состоящий из 12 элементов целого типа без знака с клавиатуры. Вывести полученный массив на экран в одну строку
3	Ввести одномерный массив, состоящий из 13 элементов целого типа со знаком с клавиатуры. Вывести полученный массив на экран в один столбец
4	Ввести одномерный массив, состоящий из 14 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в один столбец

1	2
5	Ввести одномерный массив, состоящий из 15 элементов целого типа без знака с клавиатуры. Вывести полученный массив на экран в одну строку
6	Ввести одномерный массив, состоящий из 16 элементов целого типа со знаком с клавиатуры. Вывести полученный массив на экран в один столбец
7	Ввести одномерный массив, состоящий из 11 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в одну строку
8	Ввести одномерный массив, состоящий из 12 элементов целого типа без знака с клавиатуры. Вывести полученный массив на экран в один столбец
9	Ввести одномерный массив, состоящий из 13 элементов целого типа без знака с клавиатуры. Вывести полученный массив на экран в одну строку
10	Ввести одномерный массив, состоящий из 14 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в одну строку
11	Ввести одномерный массив, состоящий из 17 элементов целого типа со знаком с клавиатуры. Вывести полученный массив на экран в один столбец
12	Ввести одномерный массив, состоящий из 18 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в одну строку
13	Ввести одномерный массив, состоящий из 10 элементов целого типа без знака с клавиатуры. Вывести полученный массив на экран в один столбец
14	Ввести одномерный массив, состоящий из 15 элементов вещественного типа с клавиатуры. Вывести полученный массив на экран в один столбец
15	Ввести одномерный массив, состоящий из 12 элементов целого типа со знаком с клавиатуры. Вывести полученный массив на экран в одну строку

Задание 2. Разработать алгоритм и программу ввода одномерного числового массива с использованием генератора случайных величин в соответствии со своим вариантом (табл. 9.2).

Т а б л и ц а 9.2

Ввод и вывод одномерного числового массива

Номер варианта	Задание
1	2
1	Ввести одномерный массив, состоящий из 18 элементов целого типа со знаком. Элементы должны находиться в диапазоне от –20 до 20. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку
2	Ввести одномерный массив, состоящий из 17 элементов вещественного типа. Элементы должны находиться в диапазоне от –22 до 22. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в один столбец
3	Ввести одномерный массив, состоящий из 18 элементов целого типа без знака. Элементы должны находиться в диапазоне от –21 до 21. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку
4	Ввести одномерный массив, состоящий из 19 элементов вещественного типа. Элементы должны находиться в диапазоне от –23 до 23. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в один столбец
5	Ввести одномерный массив, состоящий из 16 элементов целого типа со знаком. Элементы должны находиться в диапазоне от –30 до 30. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку
6	Ввести одномерный массив, состоящий из 20 элементов вещественного типа. Элементы должны находиться в диапазоне от –32 до 32. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в один столбец

Продолжение табл. 9.2

1	2
7	Ввести одномерный массив, состоящий из 15 элементов целого типа без знака. Элементы должны находиться в диапазоне от –41 до 41. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку
8	Ввести одномерный массив, состоящий из 11 элементов вещественного типа. Элементы должны находиться в диапазоне от –43 до 43. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в один столбец
9	Ввести одномерный массив, состоящий из 10 элементов целого типа со знаком. Элементы должны находиться в диапазоне от –33 до 33. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в одну строку
10	Ввести одномерный массив, состоящий из 13 элементов целого типа без знака. Элементы должны находиться в диапазоне от –45 до 45. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку
11	Ввести одномерный массив, состоящий из 20 одного элемента целого типа со знаком. Элементы должны находиться в диапазоне от –35 до 35. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в один столбец
12	Ввести одномерный массив, состоящий из 15 элементов вещественного типа. Элементы должны находиться в диапазоне от –27 до 27. При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в один столбец
13	Ввести одномерный массив, состоящий из 17 элементов целого типа без знака. Элементы должны находиться в диапазоне от –55 до 55. При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в одну строку

1	2
14	Ввести одномерный массив, состоящий из 16 элементов целого типа со знаком. Элементы должны находиться в диапазоне от $-34$ до $34$ . При каждом запуске программы элементы массива должны принимать другие значения. Вывести полученный массив на экран в один столбец
15	Ввести одномерный массив, состоящий из 13 элементов вещественного типа. Элементы должны находиться в диапазоне от $-37$ до $37$ . При каждом запуске программы элементы массива должны принимать одинаковые значения. Вывести полученный массив на экран в одну строку

Задание 3. Разработать алгоритм и программу обработки одномерных массивов в соответствии со своим вариантом (табл. 9.3). Массив  $A$  [15] задать с помощью генератора случайных величин в интервале  $[-50; 50]$ . При каждом запуске программы элементы массива должны принимать другие значения.

Таблица 9.3

## Формирование нового одномерного числового массива

Номер варианта	Задание
1	2
1	Сформировать массив $Y$ из тех элементов исходного массива $A$ , которые имеют нечетные индексы
2	Сформировать массив $V$ из тех элементов исходного массива $A$ , которые имеют в младшем разряде ноль
4	Сформировать массив $P$ из отрицательных элементов исходного массива $A$ .
5	Сформировать массив $L$ из элементов массива $A$ , больших четырех
6	Сформировать массив $X$ из тех элементов исходного массива $A$ , которые не превышают кубов своих индексов
7	Сформировать массив $U$ из тех элементов исходного массива $A$ , которые имеют четные индексы

1	2
8	Сформировать массив $D$ из элементов массива $A$ , кратных семи
9	Сформировать массив $C$ из элементов исходного массива $A$ , меньших трех
10	Сформировать массив $W$ из положительных элементов исходного массива $A$
11	Сформировать массив $R$ из четных элементов массива $A$
12	Сформировать массив $B$ из тех элементов исходного массива $A$ , которые не превышают 20
13	Сформировать массив $T$ из элементов исходного массива $A$ , кратных восьми
14	Сформировать массив $Q$ из тех элементов исходного массива $A$ , которые имеют неравные предыдущий и последующий элементы
15	Сформировать массив $K$ из элементов исходного массива $A$ , превышающих свой индекс больше чем в четыре раза

Задание 4. Разработать алгоритм и программу обработки одномерных числовых массивов в соответствии со своим вариантом (табл. 9.4). Вывести на экран исходный и результирующий массивы в две строки.

Таблица 9.4

## Обработка одномерного числового массива

Номер варианта	Задание
1	2
1	Записать на место нулевых элементов массива $D$ значение квадрата максимального элемента, подсчитать количество выполненных замен $D = \{-5; 1; 0; -15; -24; 0; 80; -23; 4; 0; 43; -7\}$
2	В массиве $X$ обнулить все элементы, расположенные между минимальным и максимальным элементами $X = \{-5; 143; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$



Продолжение табл. 9.4

1	2
3	В массиве $X$ найти наибольший элемент из элементов, удовлетворяющих условию $5 \leq X_i \leq 18$ , и его порядковый номер $X = \{-5; 6; 0; -15; -24; 10; 80; -23; 16; -31; 14; -7\}$
4	Найти в массиве $X$ максимальный элемент и заменить все элементы, стоящие после него, значением квадрата минимального элемента $X = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
5	Подсчитать количество отрицательных элементов массива $V$ , заменить их значением минимального элемента $V = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
6	В одномерном массиве $X$ обнулить все элементы, расположенные перед минимальным элементом $X = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
7	В массиве $A$ подсчитать количество нулевых элементов, заменить их значением квадрата минимального элемента $A = \{-5; 1; 0; -15; -24; 0; 80; -23; 0; -31; 43; -7\}$
8	Записать на место отрицательных элементов массива $D$ значение квадрата максимального элемента, подсчитать количество выполненных замен $D = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
9	В массиве $X$ найти минимальный из положительных элементов и заменить все элементы, расположенные после него, единицами $X = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
10	Найти в одномерном массиве $K$ максимальный элемент и заменить все элементы, стоящие после него, значением квадрата минимального элемента $K = \{-5; 1; 0; 15; -24; -75; 80; -23; 4; -31; 7\}$
11	Найти максимальное значение модуля разности между двумя соседними элементами одномерного массива $R$ и порядковые номера этих элементов $R = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
12	В массиве $X$ найти минимальный элемент среди элементов, порядковые номера которых кратны четырем $X = \{-5; 2; -1,6; 15; 0; 24,5; -75; -8; 0; 23; 72; -2\}$
13	В массиве $B$ найти наибольший элемент и заменить его значением суммы предшествующих ему элементов $B = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$
14	В массиве $G$ найти минимальный из положительных элементов и заменить все элементы, расположенные после него, единицами $G = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$

1	2
15	В массиве $N$ подсчитать сумму элементов, расположенных между минимальным и максимальным элементами $N = \{-5; 1; 0; -15; -24; -75; 80; -23; 4; -31; 43; -7\}$

Задание 5. Разработать алгоритм и программу обработки одномерных массивов в соответствии со своим вариантом (табл. 9.5). Массивы задать с помощью генератора случайных чисел в интервале  $[-100; 100]$ . При каждом запуске программы элементы массива должны принимать другие значения. Вывести на экран исходные и результирующий массивы в три строки.

Таблица 9.5

## Сортировка одномерного числового массива

Номер варианта	Задание
1	2
1	Даны два одномерных массива целых чисел, состоящих из 11 элементов. Сформировать из них один одномерный массив, состоящий из положительных элементов исходных массивов. Упорядочить полученный массив по возрастанию
2	Даны два одномерных массива целых чисел, состоящих из 12 элементов. Сформировать из них один одномерный массив, состоящий из четных элементов исходных массивов. Упорядочить полученный массив по убыванию
3	Даны два одномерных массива целых чисел, состоящих из 13 элементов. Сформировать из них один одномерный массив, состоящий из нечетных элементов исходных массивов. Упорядочить полученный массив по возрастанию
4	Даны два одномерных массива целых чисел, состоящих из 14 элементов. Сформировать из них один одномерный массив, состоящий из отрицательных элементов исходных массивов. Упорядочить полученный массив по убыванию
5	Даны два одномерных массива целых чисел, состоящих из 15 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных четырем. Упорядочить полученный массив по возрастанию

1	2
6	Даны два одномерных массива целых чисел, состоящих из 16 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных пяти. Упорядочить полученный массив по убыванию
7	Даны два одномерных массива целых чисел, состоящих из 17 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных шести. Упорядочить полученный массив по возрастанию
8	Даны два одномерных массива целых чисел, состоящих из 18 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных семи. Упорядочить полученный массив по убыванию
9	Даны два одномерных массива целых чисел, состоящих из 19 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных восьми. Упорядочить полученный массив по возрастанию
10	Даны два одномерных массива целых чисел, состоящих из 20 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных девяти. Упорядочить полученный массив по убыванию
11	Даны два одномерных массива целых чисел, состоящих из 21 элемента. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных десяти. Упорядочить полученный массив по возрастанию
12	Даны два одномерных массива целых чисел, состоящих из 22 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных шести. Упорядочить полученный массив по убыванию
13	Даны два одномерных массива целых чисел, состоящих из 23 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных пяти. Упорядочить полученный массив по возрастанию
14	Даны два одномерных массива целых чисел, состоящих из 24 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных трем. Упорядочить полученный массив по убыванию
15	Даны два одномерных массива целых чисел, состоящих из 25 элементов. Сформировать из них один одномерный массив, состоящий из элементов исходных массивов, кратных четырем. Упорядочить полученный массив по убыванию

## Библиографический список

1. Либерти Д. Освой самостоятельно С++ за 21 день / Д. Либерти. М.: Вильямс, 2015. 784 с.
2. Подбельский В. В. Язык Си++ / В. В. Подбельский. М.: Финансы и статистика, 2014. 560 с.
3. Трофимова Л. Н. Основы программирования на языке С++ / Л. Н. Трофимова, А. В. Ерошенко / Омский гос. ун-т путей сообщения. Омск, 2017. Ч. 1. 40 с.
4. Трофимова Л. Н. Основы программирования на языке С++ / Л. Н. Трофимова, А. В. Ерошенко / Омский гос. ун-т путей сообщения. Омск, 2017. Ч. 2. 25 с.

*Учебное издание*

ТРОФИМОВА Людмила Николаевна,  
ЕРОШЕНКО Александра Викторовна

## ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++

*Часть 3*

Учебно-методическое пособие

---

Редактор Н. А. Майорова

\*\*\*

Подписано в печать 21.02.2018. Формат  $60 \times 84 \frac{1}{16}$  .  
Офсетная печать. Бумага офсетная. Усл. печ. л. 1,8. Уч.-изд. л. 2,0.  
Тираж 75 экз. Заказ .

\*\*

Редакционно-издательский отдел ОмГУПСа  
Типография ОмГУПСа

\*

644046, г. Омск, пр. Маркса, 35