

# Содержание

<b>Предисловие</b>	<b>1</b>
<b>1 Лабораторные работы</b>	<b>3</b>
1.1 Массивы . . . . .	3
1.2 Файлы . . . . .	4
1.3 Структуры . . . . .	5
1.4 Обработка строк . . . . .	6
1.5 Динамические типы данных . . . . .	9
1.6 ООП . . . . .	12
<b>2 Самостоятельные работы</b>	<b>15</b>

# **Предисловие**

Предисловие

# **§ 1 Лабораторные работы**

## **§ 1.1 Массивы**

### **Лабораторная работа 1. Одномерные массивы**

1. Дан массив вещественных чисел. Найдите сумму отрицательных элементов массива.
2. Найдите произведение элементов массива с нечетными номерами.
3. Дан массив целых чисел. Количество запросить с клавиатуры. Найти максимальный (минимальный) элемент массива и его номер, при условии, что все элементы различные.
4. Найдите наименьший четный элемент массива. Если такого нет, то выведите первый элемент.
5. Преобразовать массив так, чтобы сначала шли нулевые элементы, а затем все остальные.
6. Ввести массив, в котором только два одинаковых элемента. Определить их местоположение.
7. Ввести два массива действительных чисел. Определить максимальные элементы в каждом массиве и поменять их местами.
8. Задан целочисленный массив. Определить процентное содержание элементов, превышающих среднеарифметическое всех элементов массива.
9. Выполнить сортировку массива по возрастанию (убыванию).
10. Дан массив из 10 элементов. Первые 4 упорядочить по возрастанию, последние 4 по убыванию.

### **Лабораторная работа 2. Сортировка и упорядочение массивов**

1. Создайте матрицу случайных чисел размерности  $n \times m$  в диапазоне  $[1; 10]$ .
2. Дана квадратная матрица. Вывести на экран элементы, стоящие на диагонали.

3. Дана матрица. Вывести на экран все нечетные столбцы, у которых первый элемент меньше последнего.
4. Дана матрица  $N \times M$  случайных чисел. Отсортировать элементы главной диагонали матрицы по убыванию.
5. Дана матрица  $N \times M$  случайных чисел. Упорядочить первый столбец матрицы по возрастанию, а последний столбец — по убыванию.
6. Дан массив из 10 элементов. Отсортируйте отдельно элементы от 0-го по 2-й, с 3-го по 5-й и с 6-го по 9.
7. Дан трехмерный массив  $N \times M \times K$  случайных чисел ( $N, M, K > 5$ ). Отсортируйте матрицу  $N \times M$  при  $K = 2$  и выведите её на экран монитора.
8. Дан массив 20 целых чисел на отрезке  $[-2; 5]$ . Упорядочить массив, удалив нули со сдвигом влево, ненулевыми элементами.
9. Дан массив 20 целых чисел на отрезке  $[-5; 5]$ . Упорядочить массив, удалив повторяющиеся элементы.
10. Дан массив. Найдите два соседних элемента, сумма которых минимальна.
11. В данном массиве найдите количество чисел, соседи у которых отличаются более чем в 2 раза.

## § 1.2 Файлы

### Лабораторная работа 1

1. Создайте матрицу **x[n][n]** случайных чисел. Сохраните все элементы матрицы в файл с названием **Matrix.txt**. Считайте содержимое файла **Matrix.txt** в новый массив **y[n][n]** и выведите его на экран дисплея.
2. Напишите программу, которая считывала бы элементы главной диагонали матрицы из файла **Matrix.txt**.
3. Напишите программу, которая удаляла бы  $k$ -столбец ( $1 < k < M$ ) в файле **Matrix.txt**.
4. Напишите программу, которая считывала бы элементы матрицы из файла **Matrix.txt** и записывала бы их в массив, соответствующего размера. Отсортируйте все столбцы матрицы по убыванию. Полученный массив запишите в файл **Matrix\_Sort.txt**.
5. Дан текстовый файл, содержащий целые числа. Удалить из него все четные числа.

6. В данном текстовом файле удалить все слова, которые содержат хотя бы одну цифру.
7. Напишите программу, которая считывала бы саму себя и выводила бы на экран дисплея исходный текст программы в обратном порядке.
8. Имеется файл с текстом. Осуществить шифрование данного текста в новый файл. Осуществить расшифровку полученного текста.

## § 1.3 Структуры

### Лабораторная работа 1

1. Описать структуру с именем **AEROFLOT**, содержащую следующие поля:
  - название пункта назначения рейса;
  - номер рейса;
  - тип самолета.
2. Написать программу, выполняющую следующие действия:
  - ввод с клавиатуры данных в массив, состоящий из семи элементов типа **AEROFLOT**; записи должны быть размещены в алфавитном порядке по названиям пунктов назначения (для этого выполните процедуру сортировки);
  - вывод на экран пунктов назначения и номеров рейсов, обслуживаемых самолетом, тип которого введен с клавиатуры. Если таких рейсов нет, выдать на дисплей соответствующее сообщение.
3. Описать структуру с именем **STUDENT**, содержащую следующие поля:
  - **NAME** – фамилия и инициалы;
  - **GROUP** – номер группы;
  - **SES** - успеваемость (массив из пяти элементов).
4. Написать программу, выполняющую следующие действия:
  - ввод с клавиатуры данных в массив **STUD1**, состоящий из десяти структур типа **STUDENT**; записи должны быть упорядочены по возрастанию содержимого поля **GROUP**;
  - вывод на дисплей фамилий и номеров групп для всех студентов, включенных в массив, если средний балл студента больше 4,0. Если таких нет, вывести соответствующее сообщение.

## Лабораторная работа 2

1. Информация об итогах сдачи сессии каждым студентом представлена в следующем порядке: Фамилия Имя Отчество, номер группы, экзаменационные оценки по четырём предметам. Отсортируйте фамилии студентов по алфавиту. Определить процент студентов, сдавших экзамены на 4 и 5.
2. Ведомость успеваемости студентов курса содержит следующую информацию: номер группы, фамилию, средний балл за последнюю сессию. Составить список студентов в порядке возрастания их номеров групп.
3. Даны два отсчета времени в часах, минутах и секундах. Найти величину временного интервала в секундах. Код реализовать через составной тип данных.
4. Дано пять различных дат в виде: число, месяц, год. Вывести их на экран в порядке возрастания.
5. Создать массив структур для учета занятости аудитории: день недели, время учебной пары, аудитория, название предмета. Реализовать поиск периодов времени, когда выбранная аудитория свободна.
6. Список книг содержит следующую информацию: фамилии авторов, название книги, год издания. Найти все книги, в названии которых имеется определенное слово, например, "физика".
7. Список имеющихся в продаже автомобилей содержит следующие сведения: марка автомобиля, цвет, стоимость, мощность двигателя, расход бензина на 100 км. Вывести перечень автомобилей, удовлетворяющих определенным требованиям клиента, таким например, как стоимость в диапазоне 300-500 тыс.руб., расход бензина в пределах 8-10 л и т.п.
8. Описать два комплексных числа и проделать над ними операции сложения, вычитания, умножения и деления.

## § 1.4 Обработка строк

### Лабораторная работа 1

1. В заданном тексте заменить все символы «+» на « - ». В данной задаче воспользуйтесь массивом символов (Заголовочный файл **cstring**).

```
#include<iostream>
#include<cstring>
using namespace std;
```

```

int main()
{
    char str[50]="(5+3)*7+65+7896";
    cout << "Initial string is: " << str << endl;
    for (int i=0; i<strlen(str); i++)
        if (str[i] == '+')
            str[i] = '-';
    cout << "Final string is: " << str << endl;
    return 0;
}

```

2. В данном тексте посчитать число символов «+» и «-».
3. Напишите программу, которая вычисляет длину введенной с клавиатуры строки. Реализуйте код программы, используя строковый тип данных (Заголовочный файл **string**).

```

#include <iostream>
#include <string>
using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    string s;
    cout << " Введите строку: \n"; cin >> s;
    cout << " Строка " << s << " содержит "
        << s.length() << " символов.\n";
    return 0;
}

```

4. Задана строка символов. Определить, есть ли заданный символ «э» в этой строке символов. Выведите на экран номер первого вхождения данного символа в строке..

```

#include <iostream>
#include <string>
using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    string str("Выведите на экран номер символа в строке.");
    string symbol="э";
    unsigned int pos = str.find(symbol);
    if (pos < str.length())

```

```

        cout << "Номер первого вхождения символа ("
            << symbol << "') в строке \n\n("
            << str << "')\n\n равен: " << pos << endl;
    else cout << "Такого символа в строке нет!\n";
    return 0;
}

```

5. Пусть задан некоторый текст. Вычислить, сколько раз повторяется наперед заданный символ «a».
6. Задана некоторая строка символов. Создать новую строку, которая образована из данной строки чтением от конца до начала.
7. Задано слово. Проверить, читается ли это слово слева направо и наоборот. *Простейшие слова-палиндромы:* мим, дед, наган, заказ, кабак, казак, мадам, шалаш.
8. Вводится строка слов, разделенных пробелами. Найти самое длинное слово и вывести его на экран.
9. Пусть имеется текстовый файл, содержащий несколько строк символов. Подсчитать число символов «-» в этих строках.
10. Задана строка символов. Подсчитать число слов в этой строке. Считать, что слова разделяются одним из символов « » (пробел), «,» (запятая), «.» (точка).
11. Пусть имеется текстовый файл, содержащий несколько предложений. Подсчитать количество предложений и слов в этом файле.
12. Пусть имеется текстовый файл, содержащий несколько слов. Отсортировать эти слова в алфавитном порядке и записать их в другой текстовый файл.
13. Написать программу замены данных в строке. Пусть:

**A = "123456789";**

**B = "67";**

**C = "-Шестьдесят семь-";**

Необходимо найти символы **"67"** (из строки B) и заменить их на **Шестьдесят семь-** (из строки C) в строке A, где A в итоге должна содержать **"12345-Шестьдесят семь-89"**.



## Функции работы со строками

Методы класса <b>String</b>	Описание метода
<b>s.length()</b>	Возвращает длину строки <b>s</b>
<b>s.substr(pos, length)</b>	возвращает подстроку из строки <b>s</b> , начиная с номера <b>pos</b> длиной <b>length</b> символов;
<b>s.empty()</b>	возвращает значение <b>true</b> , если строка <b>s</b> пуста, <b>false</b> — в противном случае;
<b>s.insert(pos, s1)</b>	вставляет строку <b>s1</b> в строку <b>s</b> , начиная с позиции <b>pos</b> ;
<b>s.remove(pos, length)</b>	удаляет из строки <b>s</b> подстроку <b>length</b> длиной <b>pos</b> символов;
<b>s.find(s1, pos)</b>	возвращает номер первого вхождения строки <b>s1</b> в строку <b>s</b> , поиск начинается с номера <b>pos</b> , параметр <b>pos</b> может отсутствовать, в этом случае поиск идет с начала строки;
<b>s.findfirst(s1, pos)</b>	возвращает номер первого вхождения любого символа из строки <b>s1</b> в строку <b>s</b> , поиск начинается с номера <b>pos</b> , который может отсутствовать.

Таблица 1.1: Функции работы со строками

## § 1.5 Динамические типы данных

### Лабораторная работа 1

1. Напишите программу, реализующую объявление, заполнение и удаление динамического массива. Программа также должна выполнять вывод массива на экран и запись его в текстовый (бинарный) файл.
2. Реализуйте предыдущую задачу с помощью подпрограмм (процедур и функций).
3. Дана динамическая матрица случайных чисел размерности  $N \times N$  ( $N > 9$ ). Вычислите произведение всех элементов матрицы, у которых индексы строк и столбцов четные. Результат выведите на экран.
4. Описать структуру с именем **STUDENT**, содержащую следующие поля:
  - **NAME** – фамилия и инициалы;
  - **GROUP** – номер группы;

- **SES** - успеваемость (массив из пяти элементов).

Реализовать программу, используя указатели на структуру. Запишите данные для 10 студентов в файл.

5. Создать структуру «Товар». Каждый товар должен иметь не менее 8 полей, например, название; описание; страна и город, где произведен товар; предприятие-производитель; категория товара (продукты, хозяйственные товары, промышленные товары и т.д.); цена; вес и т.д. Заполнить динамический массив десятью товарами. Реализовать поиск в массиве по названию, по вхождению слов в описание и по диапазону цены товара.
6. Объявите указатель на массив типа **double** и предложите пользователю выбрать его размер. Далее напишите четыре функции: первая должна выделить память для массива, вторая – заполнить ячейки данными, третья – показать данные на экран, четвертая – освободить занимаемую память. Для обхода массива использовать указатели (запрещено обращаться к элементам массива по индексам).

```
#include <iostream>
#include <fstream>
#include <cstdlib>

using namespace std;
int main()
{
    ofstream out("Array.txt");
    const int n=10; double* x;
    x = new double[n];
    for (int i=0; i<n; i++)
    {
        *(x+i)=rand()%10;
        cout << "x[" << i << "]=" << *(x+i) << "\t";
        out << *(x+i) << "\n";
    }
    cout << "\n"; delete[] x; out.close();

    ifstream in("Array.txt");
    double * y;
    y = new double[n];
    for (int i=0; i<n; i++)
    {
        in >> *(y+i);
        cout << "y[" << i << "]=" << *(y+i) << "\t";
    }
}
```

```

    }
    delete[] y; in.close(); return 0;
}

```

```

#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;

double* init(int n);
void data(int n, double* x);
void print(int n, double* x);
void write_file(int n, double* x);
void del(double* x);

int main()
{
    int n; cout << "Input n: "; cin >> n;
    double* x=init(n);
    data(n,x);
    print(n,x);
    write_file(n,x);
    del(x);
    return 0;
}

double* init(int n) { return new double[n]; }
void data(int n, double* x) { for (int i=0; i<n; i++) x[i]=rand()%10; }
void print(int n, double* x) { for (int i=0; i<n; i++) cout << x[i] <<
"\t"; }
void write_file(int n, double* x)
{
    ofstream out ("Array.txt");

```

```

        for (int i=0; i<n; i++) out <<"x[" << i <<"]=" << x[i] << "\n";
        out.close();
    }
    void del(double *x) { delete[] x; }

```

```

#include <iostream>
using namespace std;
int main()
{
    const int n=8, m=8;
    int **matrix;
    matrix=new int*[n];
    for (int i=0; i<n; i++)
        matrix[i]=new int[m];

    for (int i=0; i<n; i++)
    {
        for (int j=0; j<n; j++)
        {
            matrix[i][j]=(i+j); cout << matrix[i][j] << "\t";
        }
        cout << endl;
    }
    delete[] matrix;
    return 0;
}

```

## § 1.6 ООП

### Лабораторная работа 1

1. Создать класс **Employee**. Класс должен включать помимо имени и фамилии, поле типа **int** для хранения номера сотрудника и поле типа **float** для хранения величины его

оклада. Методы класса должны позволять пользователю вводить и отображать данные класса. Написать функцию **main()**, которая запросит пользователя ввести данные для трех сотрудников и выведет полученную информацию на экран.

2. Создать класс типа круг. Поля-данные: радиус, координаты центра. Функции-члены вычисляют площадь, длину окружности, устанавливают поля и возвращают значения. Функции-члены установки полей класса должны проверять корректность задаваемых параметров (не равны нулю и не отрицательные).
3. Создать класс типа время с полями: час (0–23), минуты (0–59), секунды (0–59). Класс имеет конструктор. Функции-члены установки времени, получения часа, минуты и секунды, а также две функции-члены печати: печать по шаблону «16 часов 18 минут 3 секунды» и «4 p.m. 18 минут 3 секунды». Функции-члены установки полей класса должны проверять корректность задаваемых параметров.
4. Создать класс типа дата с полями: день (1–31), месяц (1–12), год (целое число). Класс имеет конструктор. Функции-члены установки дня, месяца и года, функции-члены получения дня, месяца и года, а также две функции-члены печати: печать по шаблону «5 января 1997 года» и «05.01.1997». Функции-члены установки полей класса должны проверять корректность задаваемых параметров.
5. Создать класс одномерный массив целых чисел (вектор) с полями — количество фактических элементов, массив (динамический). Функции-члены: обращения к отдельному элементу массива, вывода массива на экран, поэлементного сложения и вычитания со скаляром, вывода элемента по заданному индексу.
6. Создать класс множество **Set**. Функции-члены реализуют добавление и удаление элемента, пересечение и разность множеств.

## Лабораторная работа 2

1. Создайте класс **Number**. Добавьте внутри класса функцию (метод) ввода переменной с клавиатуры и функцию вывода данной переменной на экран. Организуйте конструктор и деструктор с соответствующим выводом на экран сообщений «Сработал конструктор!» и «Сработал деструктор!».
2. Измените предыдущую программу таким образом, чтобы класс **Number** состоял из двух полей: целочисленной и символьной переменных. Организуйте работу деструктора и 4 конструкторов: конструктор без параметров, конструктор с целочисленным параметром, конструктор с символьным параметром, конструктор с обоими параметрами.
3. Создайте класс **Children**, который содержит такие поля (члены класса): закрытые (**private**) – имя, отчество и фамилию ребенка, а также его возраст; публичные (**public**)

– методы ввода данных и отображения их на экран. Объявить два объекта класса, внести данные и показать их. Организуйте конструктор и деструктор с соответствующим выводом на экран сообщений «**Сработал конструктор!**» и «**Сработал деструктор!**».

4. Создать класс типа параллелепипед. Поля – высота, длина и ширина. Функции-члены вычисляют площадь и объем, сумму длин всех ребер параллелепипеда и длину главной диагонали, устанавливают поля и возвращают значения. Функции-члены установки полей класса должны проверять корректность задаваемых параметров (не равны нулю и не отрицательные). Организуйте два вида конструктора: без параметров и с параметрами по умолчанию, а также деструктор с сообщением об уничтожении объекта.
5. Создайте класс «Книга», содержащий следующие поля: название, количество страниц, год издания, цена. Методы: вычисления средней стоимости страницы; сколько лет книги; определение количества дней, прошедших после года издания книги. Создайте для данного класса конструктор и деструктор.
6. Создайте класс одномерный динамический массив **Array**, который содержит такие поля (члены класса): публичные – методы ввода данных и отображения их на экран, а также определение максимального элемента массива. Создайте для данного класса конструктор и деструктор.
7. Создайте класс **Matrix**, который содержит такие поля (члены класса): публичные – методы ввода данных и отображения их на экран, а также определение максимального элемента матрицы. Создайте для данного класса конструктор и деструктор.

## **§ 2 Самостоятельные работы**