МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования

«Белорусский государственный университет транспорта»

Кафедра «Информационно-управляющие системы и технологии»

Отчет  
по практическим работам

по дисциплине «Основы алгоритмизации и программирования»

Выполнила Проверил

студентка группы ГИ-11 ст. преп. каф. «ИУСиТ»

Зайцев И. С. Голдобина Т. А.

Гомель, 2020

Оглавление

[Практическая работа № 22 Тема: «Разработка, отладка и выполнение программы обработки файлов в С++» 3](#_Toc38967962)

[1.1 Задания 3](#_Toc38967963)

[1.2 Контрольные вопросы 9](#_Toc38967964)

# Практическая работа № 22 Тема: «Разработка, отладка и выполнение программы обработки файлов в С++»

### **Цель**

Разработать, отладить и выполнить программы для обработки файлов в C++.

## Задания

### Задание 1

### Условие

Составить функцию с использование объектов cout и cin, которая запрашивает с клавиатуры 6 вещественных чисел, записывает эти числа в массив, а затем записывает их в текстовый файл tdata.

### Листинг

task1()

{

float Mass[N];

int i;

cout <<"Введите элементы массива(6)"<<endl;

for(i=0; i<N; i++)

{

cin >> Mass[i];

}

ofstream fp("tdata.txt");

if(fp == NULL)

{

cout << "Ошибка при открытии файла" << endl;

return -1;

}

for(i=0; i<N; i++)

{

fp << Mass[i] << " ";

}

fp.close();

cout <<endl;

}

### Результат

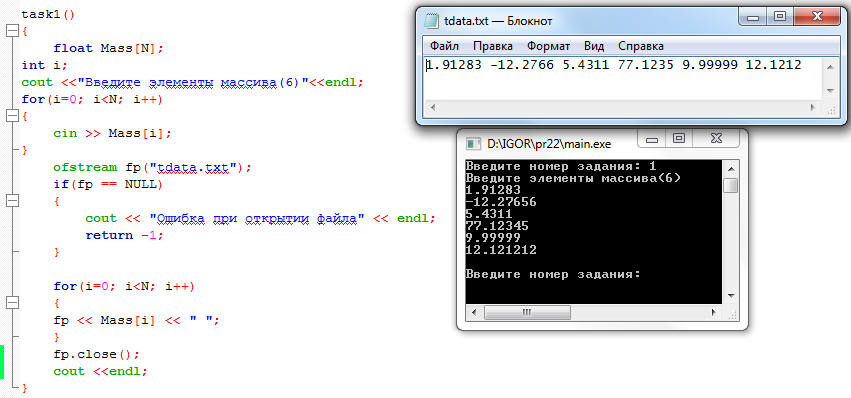


Рисунок 1 – Решение задания 1

### Задание 2

### Условие

Составить функцию, которая считывает числа из текстового файла tdata и печатает их на экране с четырьмя знаками после запятой.

### Листинг

task2()

{

float Mass[N];

int i;

ifstream fp("tdata.txt", ios::binary);

if(fp == NULL)

{

cout << "Ошибка при открытии файла" << endl;

return -1;

}

for(i=0; i<N; i++)

{

fp >> Mass[i];

}

fp.close();

for(i=0; i<N; i++)

{

cout << round(Mass[i]\*10000)/10000 <<" "; //Округление до 4 знаком после запятой

}

cout <<endl;

}

### Результат

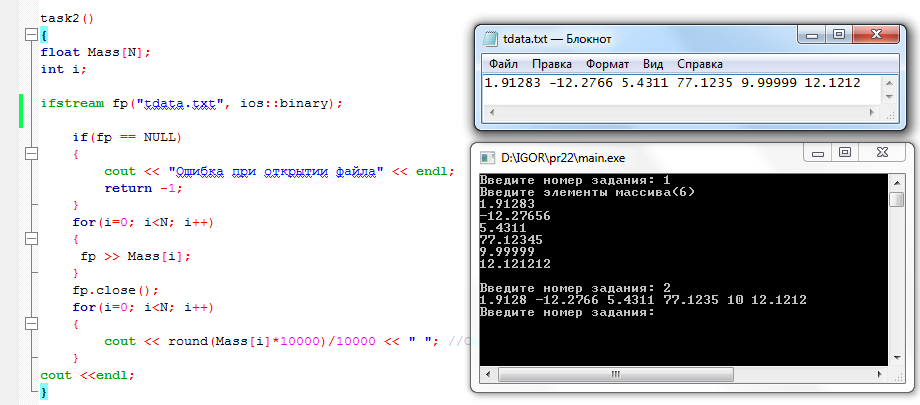


Рисунок 2 – Решение задания 2

### Задание 3

### Условие

Составить функцию, которая генерирует 31 вещественное чис(ла/ел/ло), лежащ(их/ee) в диапазоне от -23 до -4 и помещает их в массив. Записать сгенерированный массив в бинарный файл bdata.

### Листинг

task3()

{

float Mass[31];

int i, max=-4, min=-23;

for(i=0; i<31; i++)

{

Mass[i] = (float) rand()/RAND\_MAX\*(max-min)+min;

if(Mass[i]>-4){Mass[i]-=5;}

cout << Mass[i] <<" ";

}

ofstream fp("bdata.txt", ios::binary);

if(fp == NULL)

{

cout << "Ошибка при открытии файла" << endl;

return -1;

}

for(i=0; i<31; i++)

{

fp << Mass[i] << " ";

}

fp.close();

cout << endl;

}

### Результат

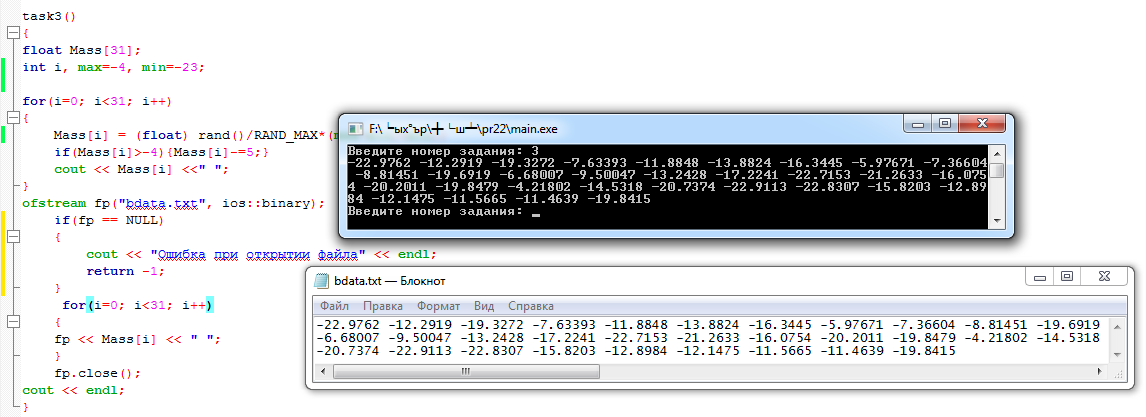


Рисунок 3 – Решение задания 3

### Задание 4

### Условие

Написать программу, которая открывает бинарный файл bdata, печатает все числа из этого файла, печатает максимальное и минимальное из значений этих чисел. Предусмотреть проверку открытия файла.

### Листинг

task4()

{

float Mass[31], max, min;

int i;

ifstream fp("bdata.txt", ios::binary);

if(fp == NULL)

{

cout << "Ошибка при открытии файла" << endl;

return -1;

}

for(i=0; i<31; i++)

{

fp >> Mass[i];

}

fp.close();

max=Mass[0];

min=Mass[0];

for(i=0;i<N; i++)

{

if(Mass[i]>max)

{

max=Mass[i];

}

if(Mass[i]<min)

{

min=Mass[i];

}

}

for(i=0; i<31; i++)

{

cout << Mass[i] <<" ";

}

cout << endl << "Максимальное число:" << max << endl << "Минимальное число:" << min <<endl;

}

### Результат

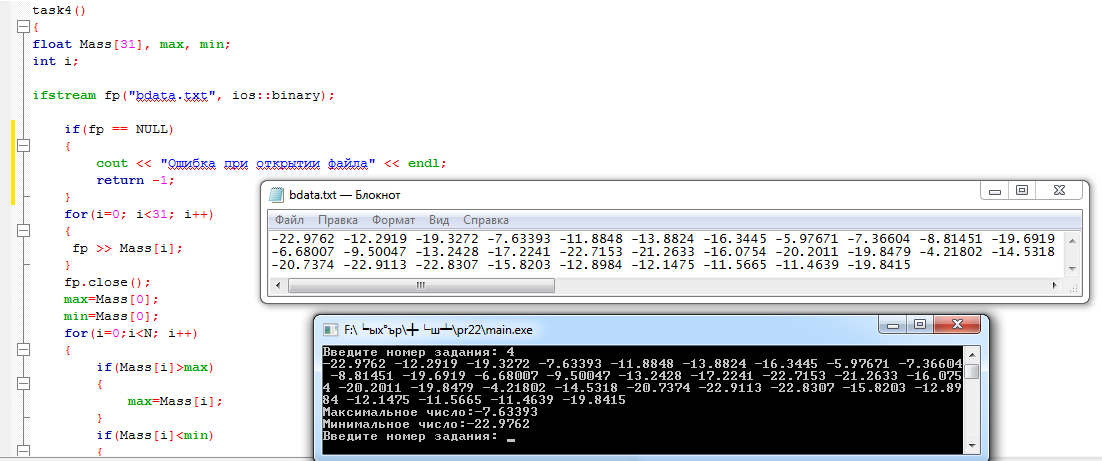


Рисунок 4 – Решение задания 4

### Задание 5

### Условие

Преобразовать программу, разработанную ранее (ПР № 11, Задание 1), заменив функции ввода/вывода языка С на функции языка С++ во всей программе. Условие задания привести полностью, включая расчетные формулы.

ПР № 11, Задание 1:

Ввести одномерный массив *U* из десяти элементов и число *n*. Для каждого элемента массива вычислить . Вывести полученный массив. Определить произведение отрицательных элементов массива *D*. Поменять местами значения последнего и минимального элементов *D*.

Вывод результатов в процессе отладки программы осуществлять на экран поэлементно в цикле.

### Листинг

task5()

{

int U[10], mini, index;

int i, D[10], n, x=1, last, z;

cout << "Введите число n(любое):";

cin >> n;

for (i=0; i<10; i++)

{

cout <<"U["<<i<<"] = ";

cin >> U[i];

}

for (i=0; i<10; i++)

{D[i]=(1/tan(U[i]))/(n+1)-log(abs(n\*U[i]));}

cout << endl;

mini=D[0];

for (i=0; i<10; i++)

{

cout <<"D["<<i<<"]="<<D[i]<<endl;

} for (i=0; i<10; i++) {

if (D[i]<0)

{

x\*=D[i];

}

if (D[i]<mini)

{

mini=D[i];

index=i;

z=D[index];

D[index]=D[9];

D[9]=z;

}

}

cout <<"Произведение элементом массива D<0: "<<x<<endl;

cout <<"После замены минимального и последнего:"<<endl;

for (i=0; i<10; i++)

{

cout <<"D["<<i<<"]="<<D[i]<<endl;

}

}

### Результат

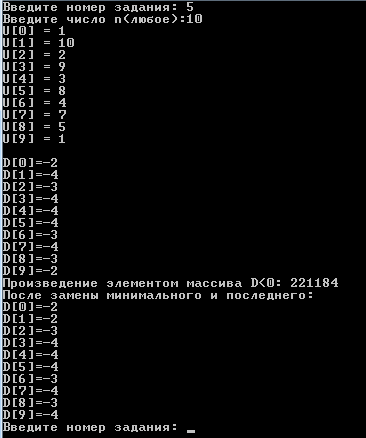


Рисунок 5 - Решение задания 5

### Задание 6

### Условие

Преобразовать программы, разработанные ранее (ПР № 15, Задания 1, 2), заменив функции ввода/вывода языка С на функции языка С++ во всей программе. Условия заданий привести полностью.

ПР № 15, Задание 1:

Составить программу, в которой вводится строка (не менее 20 символов, осмысленная, содержащая строчные и прописные буквы и цифры, которые могут быть изменены по заданию).

Выводится копия строки, в которой: сделать первые буквы каждого слова строчными, а остальные - прописными, заменить букву "к" на "т" (с учетом регистра).

ПР № 15, Задание 2:

Составить программу, в которой вводится строка (не менее 20 символов, осмысленная, содержащая строчные и прописные буквы и цифры).

Выводится строка, в которой: каждая третья буква, кроме 9-й, заменяется на прописную.

### Листинг

task6()

{

char string[1024];

char \*mass\_char;

mass\_char=string;

gets(string); /\*Hello World now 2020 god I eto karakatitsa\*/

for(int i=0; i<strlen(string); i++ ){

if(\*mass\_char == 107){\*mass\_char = 116;}

else if(\*mass\_char == 75){\*mass\_char = 84;}

if(\*mass\_char>=65 && \*mass\_char<=90){

\*mass\_char+=32;

cout << \*mass\_char;}

else if(\*mass\_char==32){

cout << \*mass\_char;

mass\_char++;

\*mass\_char=tolower(\*mass\_char);

if(\*mass\_char == 107){\*mass\_char = 116;}

cout << \*mass\_char;

i++;}

else if(\*mass\_char>=97 && \*mass\_char<=122){

\*mass\_char-=32;

cout << \*mass\_char;}

else {cout << \*mass\_char;}

mass\_char++;}

task7()

{

char string[1024];

char \*mass\_char;

mass\_char=string;

gets(string); /\*hello world now 2020 god i eto karakatitsa\*/

for(int i=2; i<7; i+=3 ){string[i]=toUpperCase(string[i]);}

for(int i=11; i<strlen(string); i+=3 ){string[i]=toUpperCase(string[i]);}

for(int i=0; i<strlen(string); i++ ){cout << string[i];}

}

char toUpperCase (char ch){

if (96<ch && ch<123 || 159<ch && ch<176){return ch-32;}

else if (223<ch && ch<240){return ch-80;}

return ch;

}

### Результат

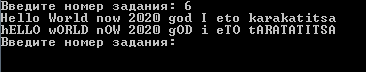


Рисунок 6.1 - Решение задания 6

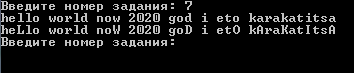


Рисунок 6.2 - Решение задания 6

## Контрольные вопросы

1. Какая библиотека С++ предназначена для потокового ввода/вывода?

#include <iostream>

1. Стандартный поток ввода

cin >>

1. Операции для работы с потоком: вставка в поток

cin >>; cout <<;

1. Манипулятор endl

endl – это манипулятор вывода, который вставляет в выходной поток символ перехода на новую строку, а затем сбрасывает буфер объекта ostream

1. Флаги форматированного ввода/вывода

right – выравнивает по правой границе;

left – выравнивает по левой границе(по умолчанию);

dec – вывод величины в десятичной системе счисления;

showpos – выводит “+” для положительных элементов;

scientific – для вещественных чисел в экспоненциальном формате;

setw(n) – задаёт ширину поля;

setprecision(n) – задает количество цифр в дробной части числа;

1. Вывод вещественных чисел с помощью флагов форматированного вывода

setprecision(n);

1. Определение количества цифр в дробной части числа с помощью функции форматирования

cout.precision(n);

1. Манипуляторы форматирования

setw(n) – задаёт ширину поля;

setprecision(n) – задает количество цифр в дробной части числа;

fixed – фиксированная форма вывода вещественных чисел;

1. Функции защиты от некорректного ввода

-;

1. Классы для работы с файлами

**Класс  ifstream;**

**Класс ofstream;**

1. Работа с текстовыми файлами на уровне потоков

#include <iostream>

#include <fstream>

int main()

{

    std::ofstream out;          // поток для записи

    out.open("D:\\hello.txt"); // окрываем файл для записи

    out.close();            // закрываем файл

    std::ifstream in;       // поток для чтения

    in.open("D:\\hello.txt"); // окрываем файл для чтения

    in.close();             // закрываем файл

    std::fstream fs;        // поток для чтения-записи

    fs.open("D:\\hello.txt"); // окрываем файл для чтения-записи

    fs.close();             // закрываем файл

    return 0;

}

### Вывод по практической работе

Разработал, отладил и выполнил программы для обработки файлов в C++.