Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа №3

По дисциплине «Основы алгоритмизации и программирования»

На тему «**Работа с файлами на языке С++**»

Выполнил:

Студент 1 курса 10 группы

Мандрик Алексей Иванович

Преподаватель: асс. Андронова М.В.

Минск, 2024

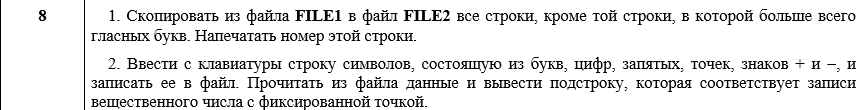
В соответствии со своим вариантом разработать программы для работы с файлами на ***языке С++.***

Для первой программы необходимо предварительно создать текстовый файл FILE1из нескольких строк и записать в него данные.

Во второй программе ввод информации с клавиатуры и вывод в консольное окно осуществить в главной функции, а запись в файл и чтение из файла − в функциях пользователя.

**Вариант 8**

**Задание 1**



**Код программы**

#include <iostream>

#include <fstream>

#include <string>

#include <cctype> // Для работы с символами

using namespace std;

// Функция для подсчета гласных букв в строке

int countVowels(const string& str)

{

int vowels = 0;

for (char ch : str)

{

if (tolower(ch) == 'a' || tolower(ch) == 'e' || tolower(ch) == 'i' || tolower(ch) == 'o' || tolower(ch) == 'u')

{

vowels++;

}

}

return vowels;

}

int main()

{

setlocale(LC\_ALL, "Rus");

ifstream inputFile("File1.txt"); // Создает объект класса ifstream с именем File1.txt

ofstream outputFile("File2.txt"); // Создает объект класса ofstream с именем File2.txt

// Если невозможно открыть

if (!inputFile || !outputFile)

{

cerr << "Ошибка при открытии файлов." << endl;

return 1;

}

string line;

string maxVowelLine;

int maxVowels = 0;

int lineNumber = 1;

int buff = 1;

while (getline(inputFile, line))

{

// Функция подсчёта гласных букв

int vowels = countVowels(line);

// Цикл для определения строки с максимальным числом гласных и определения номера этой строки

if (vowels > maxVowels)

{

maxVowels = vowels;

maxVowelLine = line;

buff = lineNumber;

}

else {

outputFile << line << endl;

}

lineNumber++;

}

cout << "Строка с наибольшим количеством гласных букв (строка " << buff << "):" << endl;

cout << maxVowelLine << endl;

//Закрытие файлов

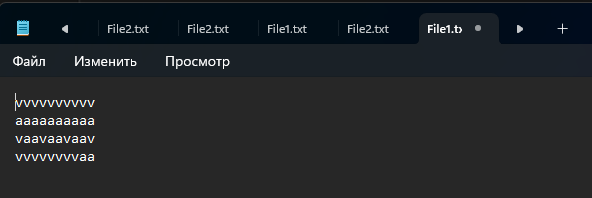
inputFile.close();

outputFile.close();

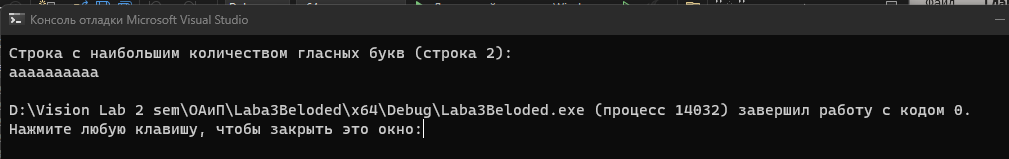
return 0;

}

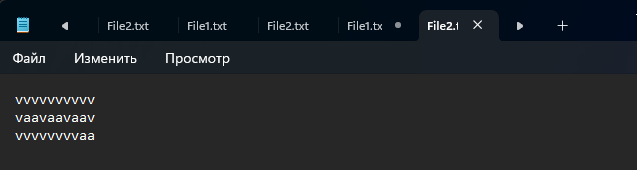
**Содержимое файла:**



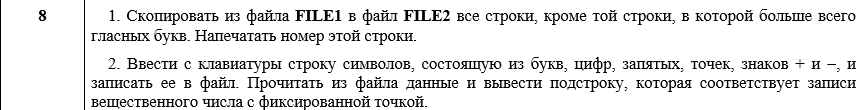
**Результат выполнения:**



**Файл после выполнения:**



**Задание 2**



**Код программы:**

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

// Функция для записи строки в файл

void write(const string& filename, const string& content)

{

ofstream outputFile(filename); // Создает объект класса ofstream с именем, передаваемым из функции

if (outputFile.is\_open())

{

outputFile << content; // Записывает строку content в файл

outputFile.close(); // Закрывает файл

cout << "Строка успешно записана в файл '" << filename << "'." << endl;

}

else

{

cerr << "Ошибка при открытии файла для записи." << endl;

}

}

// Функция для чтения данных из файла и поиска подстроки с вещественным числом

void read(const string& filename)

{

ifstream inputFile(filename); // Создает объект класса ifstream с именем, передаваемым из функции

if (inputFile.is\_open())

{

string line;

while ( getline(inputFile, line))

{

// Поиск подстроки, соответствующей записи вещественного числа с фиксированной точкой

size\_t startPos = line.find\_first\_of("0123456789");

size\_t endPos = line.find\_last\_of("0123456789");

if (startPos != string::npos && endPos != string::npos)

{

string substring = line.substr(startPos, endPos - startPos + 1);

cout << "Найденная подстрока: " << substring << endl;

}

else

{

cout << "Подстрока не найдена." << endl;

}

}

inputFile.close();

}

else

{

cerr << "Ошибка при открытии файла для чтения." << endl;

}

}

int main()

{

setlocale(LC\_ALL, "Rus");

// Ввод строки с клавиатуры

cout << "Введите строку: ";

string inputString;

getline( cin, inputString);

// Функция записи строки в файл

write("File.txt", inputString);

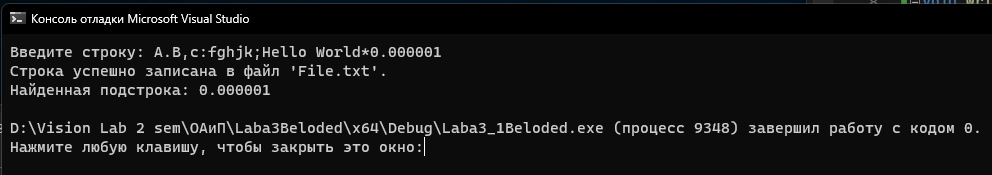
// Функция чтения данных из файла

read("File.txt");

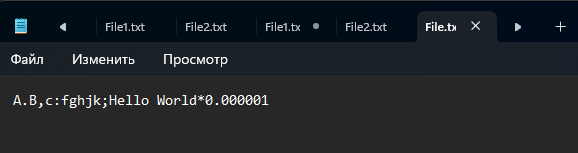
return 0;

}

**Результат выполнени:**

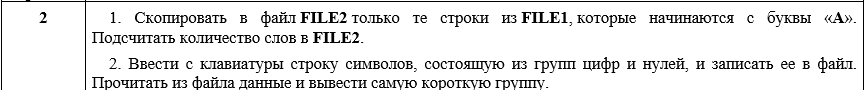


**Содержимое файла после выполнения:**



**Дополнительные задания:**

**Вариант 2**

****

**Код программы:**

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

string str;

// Открытие файлов для чтения/записи

fstream inp("File1.txt", ios::in);

fstream out("File2.txt", ios::out);

int space\_count = 0;

bool is\_space = false;

// Цикл для определния нужных строк

while (getline(inp, str))

{

// Если первый символ строки равен 'A', то:

if (str.front() == 'A')

{

is\_space = false;

// Записываем эту строку в "File2.txt".

out << str << "\n";

for (int i = 0; i < str.length(); i++)

{

// Подсчитываем количество слов.

if (str[i] == ' ')

{

space\_count++;

is\_space = true;

}

}

if (!is\_space)

{

space\_count++;

}

}

}

space\_count++;

std::cout << "Количество слов в файле File2.txt = " << space\_count << endl;

// Закрытие файлов

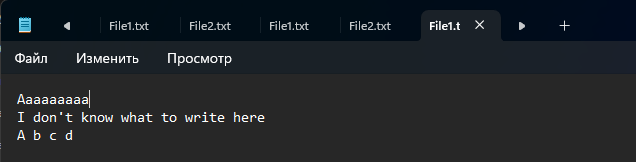
inp.close();

out.close();

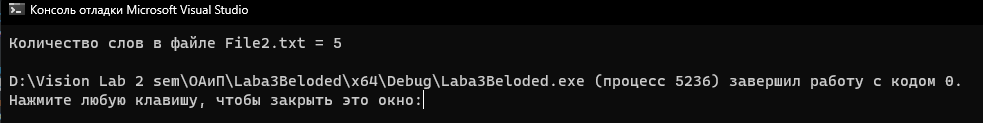
return 0;

}

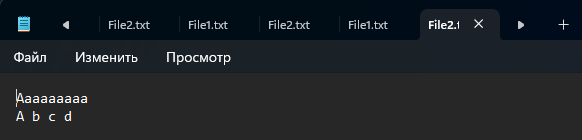
**Содержимое файла:**



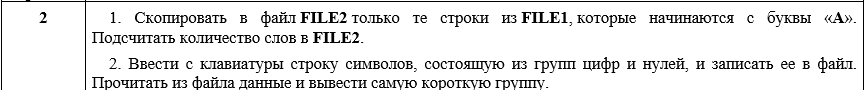
**Результат выполнения:**



**Содержимое файла после:**



**Задание 2**

****

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

char inputString[256];

cout << "Введите строку: ";

cin.getline(inputString, 256);

// Запись строки в файл

fstream outFile("File1.txt", ios::out);

outFile << inputString;

outFile.close();

// Чтение данных из файла

fstream inFile("File1.txt", ios::in);

string fileString;

getline(inFile, fileString);

inFile.close();

// Поиск самой короткой группы

int minGroupLength = 1000000; // Большое начальное значение

int groupLength = 0;

int minGroupStart = 0;

for (int i = 0; i < fileString.length(); i++) {

if (fileString[i] == '0' || fileString[i] == '1') {

groupLength++;

}

else {

if (groupLength > 0 && groupLength < minGroupLength) {

minGroupLength = groupLength;

minGroupStart = i - groupLength;

}

groupLength = 0;

}

}

// Вывод

if (minGroupLength > 0) {

cout << "Самая короткая группа: ";

for (int i = minGroupStart; i < minGroupStart + minGroupLength; i++) {

cout << fileString[i];

}

cout << endl;

}

else {

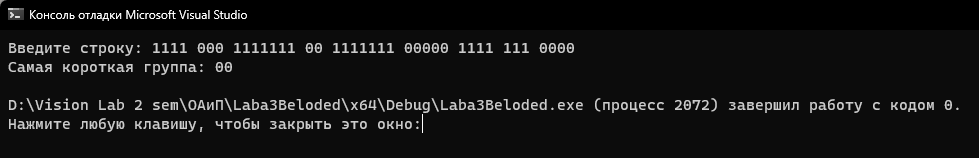
cout << "В строке нет групп нулей или единиц." << endl;

}

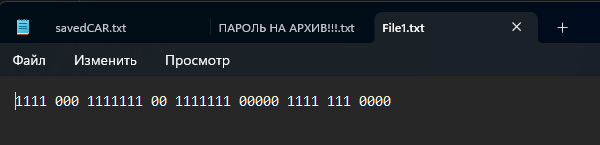
return 0;

}

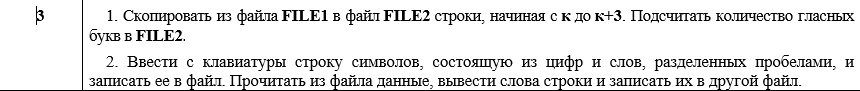
**Результат выполнения:**

****

**Содержимое файла:**

****

**Вариант 3**

****

**Код программы:**

#include <iostream>

#include <fstream>

#include <string>

#include <cctype> // Для функции isalpha()

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

int k = 2; // Переменная

// Открыть файл для записи/чтения

fstream inFile("File1.txt", ios::in);

fstream outFile("File2.txt", ios::out);

// Скопировать строки из FILE1 в FILE2

string line;

int currentLine = 0;

while (getline(inFile, line))

{

if (currentLine >= k - 1 && currentLine < k + 3)

{

outFile << line << endl;

}

currentLine++;

}

// Закрыть

inFile.close();

outFile.close();

// Подсчет количества гласных букв в File2

fstream countFile("File2.txt", ios::in);

string fileString;

string fullFileContent;

while (getline(countFile, fileString))

{

fullFileContent += fileString;

}

countFile.close();

int vowelCount = 0;

for (char c : fullFileContent)

{

if (isalpha(c))

{

char lowercaseC = tolower(c);

if (lowercaseC == 'a' || lowercaseC == 'e' || lowercaseC == 'i' || lowercaseC == 'o' || lowercaseC == 'u' || lowercaseC == 'y')

{

vowelCount++;

}

}

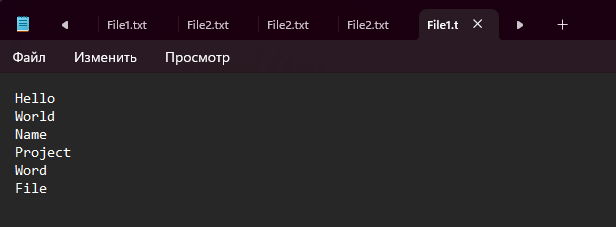
}

cout << "Количество гласных букв в FILE2: " << vowelCount << endl;

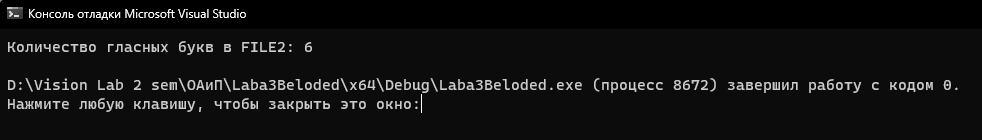
return 0;

}

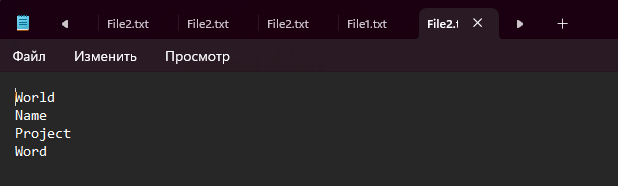
**Содержимое файла:**

****

**Результат выполнения:**

****

**Содержимое файла после выполнения:**

****

**Задание 2**

**Код программы:**

#include <iostream>

#include <fstream>

#include <sstream>

#include <string>

using namespace std;

int main() {

string inputString;

cout << "Введите строку: ";

getline(cin, inputString);

// Запись в файл File1.txt

ofstream outputFile1("File1.txt");

if (outputFile1.is\_open()) {

outputFile1 << inputString;

outputFile1.close();

cout << "Строка успешно записана в файл File1.txt." << endl;

}

else {

cout << "Не удалось открыть файл для записи." << endl;

}

// Чтение из файла File1.txt и запись в файл File2.txt

ifstream inputFile("File1.txt");

ofstream outputFile2("File2.txt");

if (inputFile.is\_open() && outputFile2.is\_open()) {

string inputString;

getline(inputFile, inputString);

istringstream iss(inputString);

string word;

while (iss >> word) {

// Проверка, что слово состоит только из букв

bool isWord = true;

for (char c : word) {

if (!isalpha(c)) {

isWord = false;

break;

}

}

if (isWord) {

outputFile2 << word << endl;

}

}

cout << "Слова успешно записаны в файл File2.txt." << endl;

inputFile.close();

outputFile2.close();

}

else {

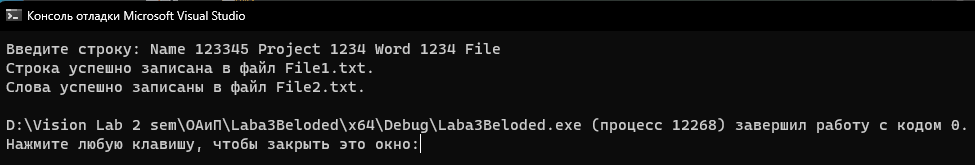
cout << "Не удалось открыть файлы для чтения или записи." << endl;

}

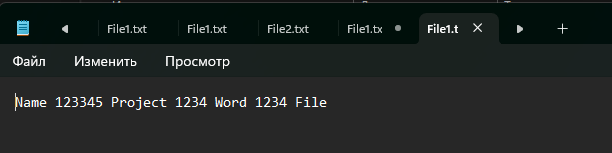
return 0;

}

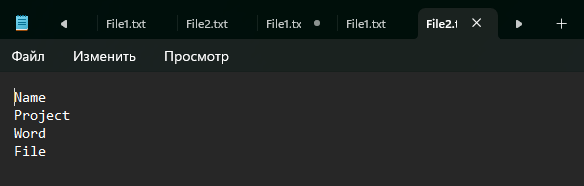
**Результат выполнения:**

****

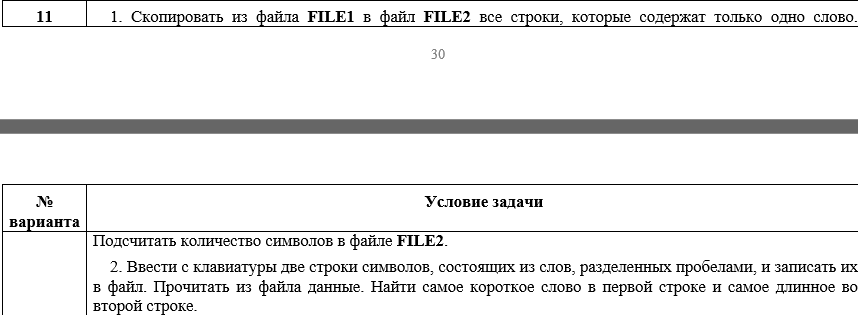
**Содержимое File1.txt;**

****

**Содержимое File2.txt;**

****

**Вариант 11**

****

**Задание 1**

**Код программы:**

#include <iostream>

#include <fstream>

#include <sstream>

#include <string>

using namespace std;

int main() {

setlocale(LC\_ALL, "Rus");

string str;

int numberStr, count = 0, min = 100;

ofstream ofile\_1("File1.txt"); //Открыть

cout << "Введите кол-во строк: "; // Количество вводимых строк

cin >> numberStr;

// Запись строк в файл

for (int i = 0; i < numberStr + 1; i++)

{

getline(cin, str);

ofile\_1 << str << endl;

}

//Закрыть

ofile\_1.close();

// Открыть для записи/чтения

ifstream ifile\_1("File1.txt");

ofstream ofile\_2("File2.txt");

cout << "Сторки, которые содержат только одно слово: ";

// Цикл определения количества слов

for (int i = 0; i < numberStr + 1; i++)

{

count = 0;

getline(ifile\_1, str);

for (int j = 0; str[j] != '\0'; j++)

if (str[j] == ' ') count++;

// Запись в File2.txt строк с одним словом

if (!(count)) {

ofile\_2 << str << endl;

cout << str << endl;

}

}

// Закрыть

ifile\_1.close();

ofile\_2.close();

string buffStr;

ifstream ifile\_2("File2.txt");// Отркыть для чтения

//Цикл определения самого короткого слова

for (int i = 0; i < numberStr + 1; i++)

{

count = 0;

getline(ifile\_2, str);

count = str.size();

if (min > count && count != 0) {

min = count;

buffStr = str;

}

}

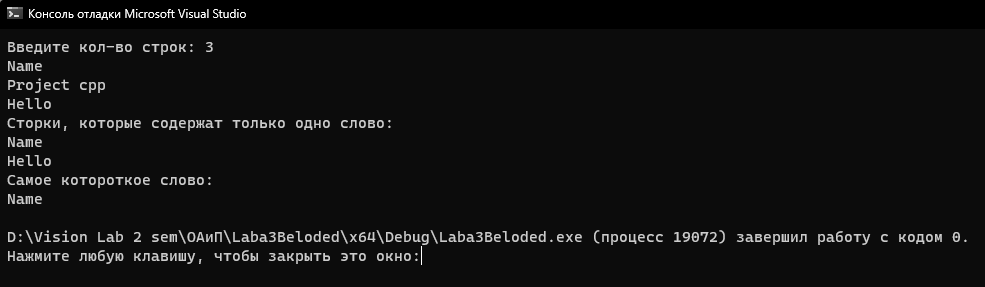
cout << "Самое котороткое слово: \n";

cout << buffStr << endl;

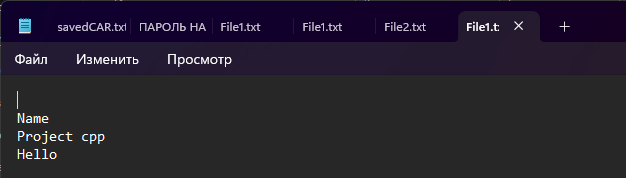
return 0;

}

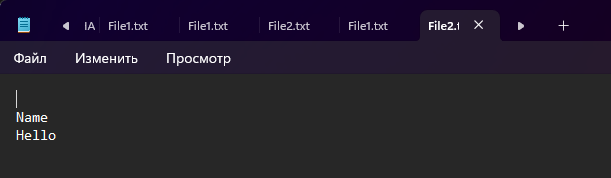
**Результат выполнения:**

****

**Содержимое файла File1.txt:**

****

**Содержимое файла File2.txt:**

****

**Задание 2**

**Код программы:**

#include <iostream>

#include <fstream>

#include <sstream>

#include <string>

using namespace std;

int main() {

setlocale(LC\_ALL, "Rus");

string str, buffStrMin, buffStrMax, buff;

int numberStr = 2, count = 0, min = 100, max = 0, j = 0;

ofstream ofile\_1("File1.txt"); //Открыть для записи

cout << "Введите две строки строки: ";

for (int i = 0; i < numberStr; i++)

{

getline(cin, str);

ofile\_1 << str << endl;

}

// Закрыть

ofile\_1.close();

// Открыть для чтения

ifstream ifile\_1("File1.txt");

for (int i = 0; i < numberStr; i++) {

getline(ifile\_1, str);

string word; //Буфер для считывания строки

stringstream stream; //Создание потоковой переменной

stream << str; //Перенос строки в поток

// Определения минимального и максимального слова

while (stream >> word) {

count = word.size();

if (!i && min > count) {

min = count;

buffStrMin = word;

}

if (i && max < count) {

max = count;

buffStrMax = word;

}

}

}

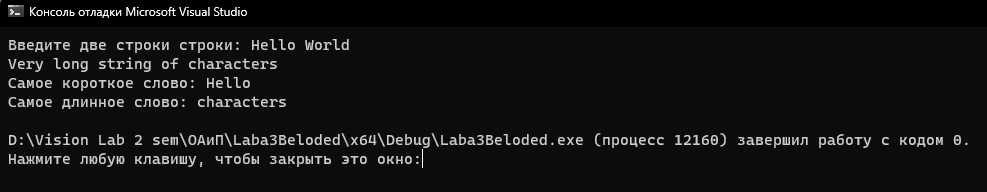
cout << "Самое короткое слово: " << buffStrMin << endl;

cout << "Самое длинное слово: " << buffStrMax << endl;

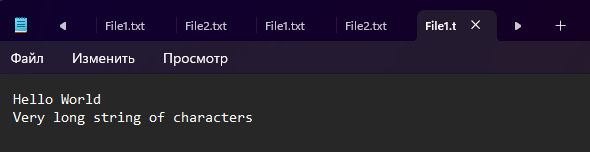
return 0;

}

**Результат выполнения:**

****

**Содержимое файла:**

****