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++»

 Выполнила:

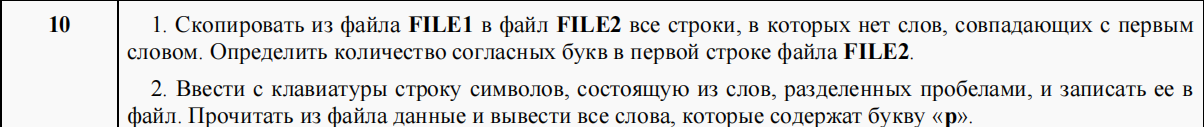
Студентка1 курса 6 группы

Литвинчук Дарья Валерьевна

Преподаватель: асс. Андронова М.В.

2023, Минск

**Вариант 10 (основной)**



1.

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

bool isVow(char\* symb) // функция для проверки является ли символ каким-то кроме согласной

{

char s = \*symb;

if (s == 'a' || s == 'e' || s == 'i' || s == 'o' || s == 'u' || s == 'A' || s == 'E' || s == 'I' || s == 'O' || s == 'U' || s == ' ' || s == '.' || s == ',') // проверяем

return true; // возвращаем true

else

return false; // возвращаем false

}

bool checkLine(const string& line, const string& word) // функция для проверки строки на наличие первого слова

{

int wordLength = word.length(); // определяем длину первого слова

int lineLength = line.length(); // определяем длину строки

for (int i = 0; i <= lineLength; i++) // цикл для того чтобы бежать по строке

{

if (line[i] == word[0]) // если первый символ слова совпадает с текущим симолом в строке, то

{

int j;

for (j = 0; j < wordLength; j++) // цикл для проверки совпадает ли первое слово со словом в строке

{

if (line[i + j] != word[j]) // если условие не выполниться, выходим из цикла

{

break;

}

}

if (i == 0 || line[i - 1] == ' ' && line[i + wordLength] == ' ') // если слово полностью подходит

{

return true; // возвращаем true

}

}

}

return false; // если в предыдущем цикле мы не вышли из функции значит совпадений нет, поэтому возвращаем false

}

int main()

{

setlocale(LC\_ALL, "rus");

ifstream file1("FILE1.txt"); // октрываем файл для чтения

ofstream file2("FILE2.txt"); // открываем файл для записи

string line, firstWord;

if (file1.fail()) // проверка на открытие файла

{

cout << "\n Ошибка открытия FILE1.txt";

return 1;

}

if (file2.fail()) // проверка на открытие файла

{

cout << "\n Ошибка открытия FILE2.txt";

return 1;

}

file1 >> firstWord; // записываем в переменную первое слово

cout << "первое слово: " << firstWord << endl; // выводим первое слово

file1.clear();// возвращаем позицию чтения файла на начало

file1.seekg(0);

bool f = true; // переменная для того чтобы мы остановили подсчёт согласных после первой строки

int count = 0; // создаём переменную для подсчёта согласных букв тексте

while (getline(file1, line)) // цикл для пробега по всем строкам из файла

{

if (f) // если f=true

{

for (int i = 0; line[i] != '\0'; i++) // бежим по всей строке

{

if (!isVow(&line[i])) // если текущий символ не является гласным или спец символом

{

count++; //увеличиваем количество согласных

}

}

f = false; // присваивем false чтобы больше не считать согласные в других строках

}

if (!checkLine(line, firstWord)) // вызваем функцию и если выполняется условие

{

file2 << line << "\n"; // то записываем строку в файл

}

}

cout << "Количество согласных в первой строке: " << count; // выводим количество согласных в консоль

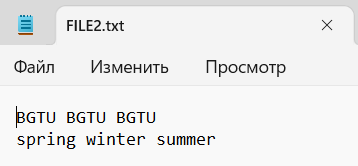
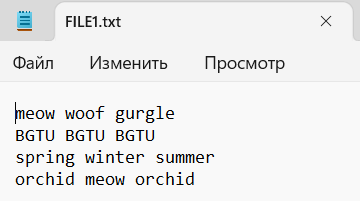
//закрываем файлы

file1.close();

file2.close();

return 0;

}





2.

#include <fstream>

#include <iostream>

using namespace std;

void main()

{

setlocale(LC\_ALL, "rus");

char buff[100];// буфер для хранения считываемого из файла текста

ofstream fout("t.txt");// создание объекта fout класса ofstream для записи

fout << "Январь Февраль Март Апрель Май Июнь Июль Август Сентябрь Октябрь Ноябрь Декабрь"; // запись строку в файл

fout.close();

ifstream fin("t.txt");// создание объекта fin класса ifstream для чтения

if (!fin.is\_open())

cout << "Файл не может быть открыт!\n";

else

{

cout << "Слова с буквой 'p' в строке: "; // вывод соотв. сообщения

while (fin >> buff) { // выводим слова в поток

for (int i = 0; i < strlen(buff); i++) { // посимвольно проверяем выведенное слово

if (buff[i] == 'р') // если есть "p", то выводим слово

cout << buff << " ";

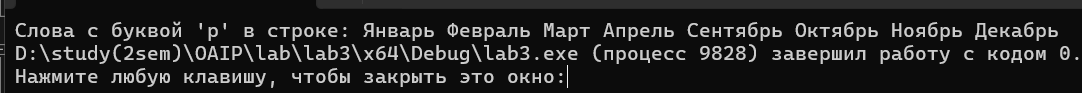
}

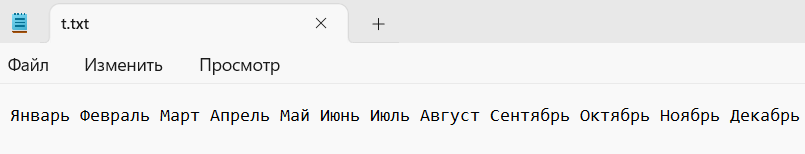
}

}

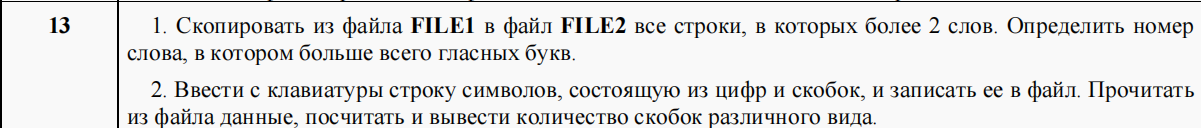
fin.close();

}





**Вариант 13**



1.

#include <fstream>

#include <iostream>

#include <cstring>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

char buff[50]; // буфер для хранения считываемого из файла текста

ofstream fout1("FILE1.txt"), fout2("FILE2.txt"); // создание объектов fout класса ofstream для записи

fout1 << "park\n"; // запись строки в файл

fout1 << "how much wood would a woodchuck chuck \n";

fout1 << "red wood maple\n";

fout1 << "woodland \n";

fout1 << "forest red";

fout1.close();

ifstream fin1("FILE1.txt"), fin2("FILE2.txt"); // создание объектов fin класса ifstream для чтения

if (!fin1.is\_open() || !fin2.is\_open()) {

cout << "Файл не может быть открыт!\n";

return 1;

}

while (fin1.getline(buff, sizeof(buff))) {

int counter\_words = 1; // по умолчанию считается, что в строке 1 слово

for (int i = 0; buff[i] != '\0'; i++) {

if (buff[i] == ' ' && buff[i + 1] != '\0') { //если после пробела не \0

counter\_words++; // то увеличить счетчик

}

}

if (counter\_words > 2) { //если число слов > 2

fout2 << buff << endl; //добавить его в файл 2

}

}

fin1.close(); //закрытие файлов

fout2.close();

int counter\_vowels = 0, max\_vowels = 0, i = 0;

char vword[50];

fin2.getline(buff, sizeof(buff)); //п

while (buff[i] != '\0' && buff[i] != -52) {

int index = i; //переопределяем позицию слова

int length = 0; // переопределяем длину слова

counter\_vowels = 0; //обнуляем счетчик гласных каждую итерацию

while (buff[i] != ' ') { //пока слово не закончилось

length++; //увеличиваем его длину

if (buff[i] == 'e' || buff[i] == 'u' || buff[i] == 'o' || buff[i] == 'i' || buff[i] == 'a') { //если симвод - гласная

counter\_vowels++; //увеличить счетчик

}

i++;

}

if (counter\_vowels > max\_vowels) {

max\_vowels = counter\_vowels; //если счетчик макс. гласных переопределен

for (int j = 0; j < length; j++) {

vword[j] = buff[index++]; //перезаписываем искомое слово

}

vword[length] = '\0'; //устанавливаем нуль-символ в конец строки

}

i++;

}

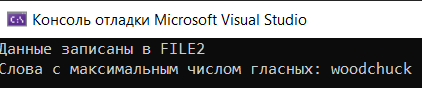
fin2.close();

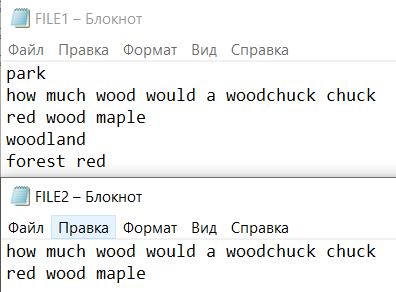
cout << "Данные записаны в FILE2 " << endl; //вывод соответствующего сообщения

cout << "Слова с максимальным числом гласных: " << vword << endl; //вывод слова с максимальным количеством гласных во втором файле

return 0;

}





2.

#include <iostream>

#include <fstream>

#include <string>

#include <cctype>

using namespace std;

void inFile(const char str[]); // функция записи строки в файл

void fromFile(); // функция чтения из файла

int main() {

setlocale(LC\_ALL, "Russian");

char str[100];

cout << "Введите строку из цифр и скобок: \n"; // ввод строки с клавиатуры

cin.getline(str, sizeof(str));

inFile(str);

fromFile();

cout << endl;

return 0;

}

// для записи строки в файл

void inFile(const char str[]) {

ofstream ofile("test.txt"); // создание объекта класса ofstream для записи

if (ofile.fail()) // проверка открытия файла

{

cout << "Ошибка открытия файла\n ";

exit(1);

}

ofile << str; // запись строки в файл

ofile.close();

cout << "Строка записана в файл\n";

}

// для чтения различных из файла

void fromFile() {

ifstream ifile("test.txt"); // создание объекта класса ifstream для чтения

if (ifile.is\_open()) {

char output[100];

int braces\_op = 0, braces\_cl = 0, square\_op = 0, square\_cl = 0, round\_op = 0, round\_cl = 0; // счетчики скобок

ifile.getline(output, sizeof(output));

for (int i = 0; i < sizeof(output); i++) { //посимвольно проверяем скобки

if (output[i] == '{') {

braces\_op++;

}

if (output[i] == '}') {

braces\_cl++;

}

if (output[i] == '[') {

square\_op++;

}

if (output[i] == ']') {

square\_cl++;

}

if (output[i] == '(') {

round\_op++;

}

if (output[i] == ')') {

round\_cl++;

}

}

//выводим соответствующие значения

cout << "Количетсво { : " << braces\_op << endl;

cout << "Количетсво } : " << braces\_cl << endl;

cout << "Количетсво [ : " << square\_op << endl;

cout << "Количетсво ] : " << square\_cl << endl;

cout << "Количетсво ( : " << round\_op << endl;

cout << "Количетсво ) : " << round\_cl << endl;

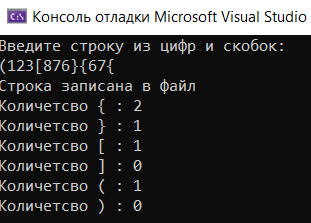
}

else {

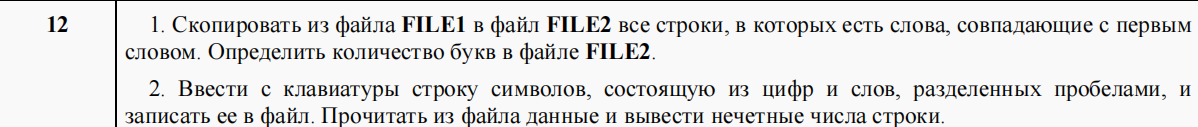
"\nОшибка открытия файла";

}

}



**Вариант 12**

1.

#include <fstream>

#include <iostream>

#include <cstring>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

char buff[50], first\_word[50]; // буфер для хранения считываемого из файла текста и первого слова из файла

ofstream fout1("FILE1.txt"), fout2("FILE2.txt"); // создание объектов fout класса ofstream для записи

fout1 << "wood park\n"; // запись строки в файл

fout1 << "how much wood would a woodchuck chuck \n";

fout1 << "red wood \n";

fout1 << "forest red";

fout1.close();

ifstream fin1("FILE1.txt"), fin2("FILE2.txt"); // создание объектов fin класса ifstream для чтения

if (!fin1.is\_open() || !fin2.is\_open()) {

cout << "Файл не может быть открыт!\n";

return 1;

}

fin1 >> first\_word; // считывание первого слова из файла

int counter\_letters = 0; // переменная для подсчёта букв во втором файле

while (fin1.getline(buff, sizeof(buff))) {

if (strstr(buff, first\_word)) { // если первое слово есть в строке,

fout2 << buff << endl; // то добавляем его во второй файл

for (int i = 0; buff[i] != '\0'; i++) { // подсчёт количества букв в строке

if (isalpha(buff[i])) {

counter\_letters++;

}

}

}

}

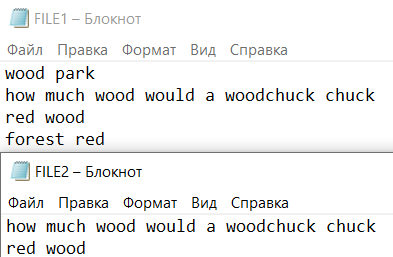
fin1.close(); //закрытие файлов

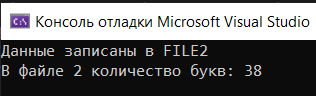
fout2.close();

cout << "Данные записаны в FILE2 " << endl; //вывод соответствующего сообщения

cout << "В файле 2 количество букв: " << counter\_letters << endl; //вывод количества букв во втором файле

}





2.

#include <iostream>

#include <fstream>

#include <string>

#include <cctype>

using namespace std;

void inFile(const char str[]); // функция записи строки в файл

void fromFile(); // функция чтения из файла

int main() {

setlocale(LC\_ALL, "Russian");

char str[100];

cout << "Введите строку из цифр и слов: \n"; // ввод строки с клавиатуры

cin.getline(str, sizeof(str));

inFile(str);

fromFile();

cout << endl;

return 0;

}

// для записи строки в файл

void inFile(const char str[]) {

ofstream ofile("test.txt"); // создание объекта класса ofstream для записи

if (ofile.fail()) // проверка открытия файла

{

cout << "Ошибка открытия файла\n ";

exit(1);

}

ofile << str; // запись строки в файл

ofile.close();

cout << "Строка записана в файл\n";

}

// для чтения слов с буквой "p" из файла

void fromFile() {

ifstream ifile("test.txt"); // создание объекта класса ifstream для чтения

if (ifile.is\_open()) {

char output[100];

cout << "Нечетные числа в строке: "; // вывод соотв. сообщения

while (ifile >> output) { // выводим значение в поток

if (isdigit(\*output)) { //если это число

int number = stoi(output); //преобразовываем его в число

if (number % 2 != 0) { // проверяем его на нечетность

cout << number << " "; //выводим в случае истины

}

}

}

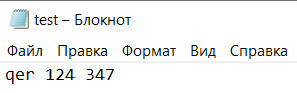
}

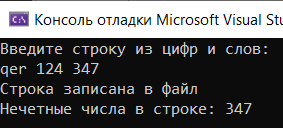
else {

"\nОшибка открытия файла";

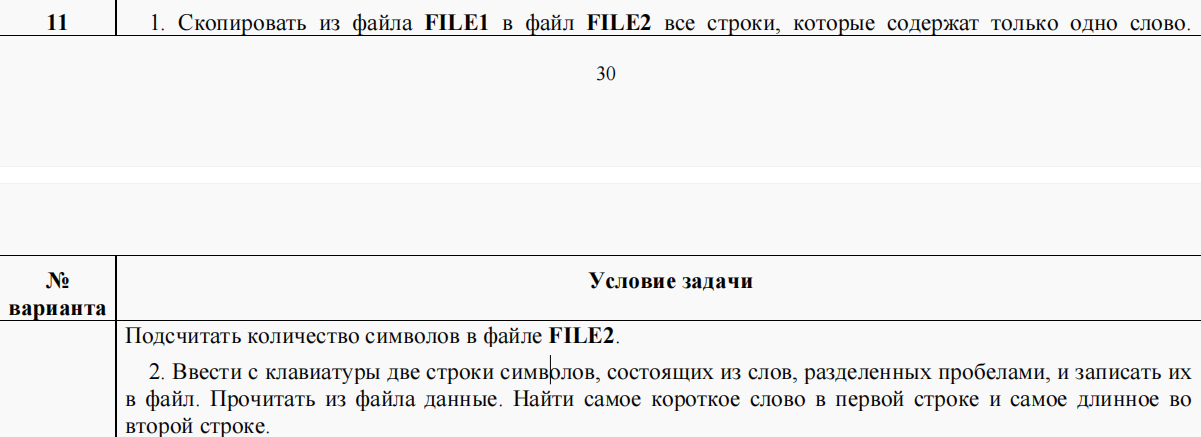
}

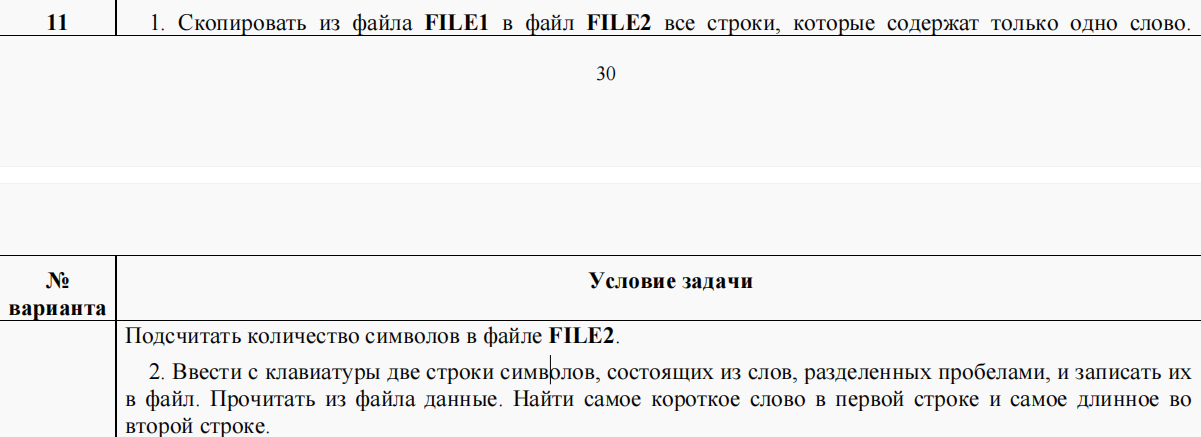
}





**Вариант 11**





1.

#include <fstream>

#include <iostream>

#include <cstring>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

char buff[50]; // буфер для хранения считываемого из файла текста

ofstream fout1("FILE1.txt"), fout2("FILE2.txt"); // создание объектов fout класса ofstream для записи

fout1 << "park\n"; // запись строки в файл

fout1 << "how much wood would a woodchuck chuck \n";

fout1 << "red wood\n";

fout1 << "woodland \n";

fout1 << "forest red";

fout1.close();

ifstream fin1("FILE1.txt"), fin2("FILE2.txt"); // создание объектов fin класса ifstream для чтения

if (!fin1.is\_open() || !fin2.is\_open()) {

cout << "Файл не может быть открыт!\n";

return 1;

}

int counter\_symbols = 0; // переменная для подсчёта символов во втором файле

while (fin1.getline(buff, sizeof(buff))) {

int counter\_words = 1; // по умолчанию считается, что в строке 1 слово

for (int i = 0; buff[i] != '\0'; i++) {

if (buff[i] == ' ' && buff[i + 1] != '\0') { //если после пробела не \0

counter\_words++; // то увеличить счетчик

}

}

if (counter\_words == 1) { //если число слов = 1

fout2 << buff << endl; //добавить его в файл 2

counter\_symbols += strlen(buff); // расчет количества символов во 2 файле

}

}

fin1.close(); //закрытие файлов

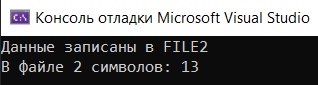
fout2.close();

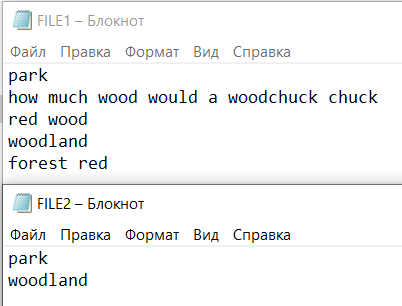
cout << "Данные записаны в FILE2 " << endl; //вывод соответствующего сообщения

cout << "В файле 2 символов: " << counter\_symbols << endl; //вывод количества символов во втором файле

return 0;

}





2.

#include <iostream>

#include <fstream>

#include <string>

#include <cctype>

using namespace std;

void inFile(const char str[], const char str2[]); // функция записи строки в файл

void fromFile(); // функция чтения из файла

int main() {

setlocale(LC\_ALL, "Russian");

char str1[100], str2[100];

cout << "Введите строку 1 \n"; // ввод строки 1 с клавиатуры

cin.getline(str1, sizeof(str1));

cout << "Введите строку 2 \n"; // ввод строки 2 с клавиатуры

cin.getline(str2, sizeof(str2));

inFile(str1, str2);

fromFile();

cout << endl;

return 0;

}

// для записи строки в файл

void inFile(const char str1[], const char str2[]) {

ofstream ofile("test.txt"); // создание объекта класса ofstream для записи

if (ofile.fail()) // проверка открытия файла

{

cout << "Ошибка открытия файла\n ";

exit(1);

}

ofile << str1 << endl; // запись строки в файл

ofile << str2;

ofile.close();

cout << "Строка записана в файл\n";

}

// для чтения различных из файла

void fromFile() {

ifstream ifile("test.txt"); // создание объекта класса ifstream для чтения

if (ifile.is\_open()) {

char line1[100], line2[100];

ifile.getline(line1, sizeof(line1)); //получение первой строки

int i = 0, imin = 100;

char shortest\_word[100]; //короткое слово в строке

while (line1[i] != '\0') { // пока не конец строки

int index = i; //устанавливаем точку отсчета начало слова

int length = 0; //длина слова

while (line1[i] != ' ' && line1[i] != '\0') { //пока слово не закончится

length++; //увеличиваем его длину

i++;

}

if (length < imin) { //если минимальный размер переопределен

imin = length;

for (int j = 0; j < imin; j++) { //меняем короткое слово по индексу

shortest\_word[j] = line1[index++];

}

shortest\_word[length] = '\0'; // устанавливаем нуль-символ в конец слова

}

i++;

}

ifile.getline(line2, sizeof(line2)); //получение второй строки

char longest\_word[100]; // длинное слово в строке

int imax = 0; //максимальный размер слова в строке

i = 0; // переопределние счетчика

while (line2[i] != '\0' && line2[i] != -52) { // пока не конец строки

int index = i; //устанавливаем точку отсчета начало слова

int length = 0; //длина слова

while (line2[i] != ' ' && line2[i] != '\0') { //пока слово не закончится

length++; //увеличиваем его длину

i++;

}

if (length > imax) { // если макс. размер переопределен

imax = length;

for (int j = 0; j < imax; j++) { //меняем длинное слово по индексу

longest\_word[j] = line2[index++];

}

longest\_word[length] = '\0'; // устанавливаем нуль-символ в конец слова

}

i++;

}

cout << "Самое короткое слово из первой строки: " << shortest\_word << endl;

cout << "Самое длинное слово из второй строки: " << longest\_word;

}

else {

cout << "Ошибка открытия файла\n";

}

}

