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(а):

Студент(ка) 1 курса 8 группы

Семёнов Даниил Вячеславович

Преподаватель: асс. Андронова М.В.

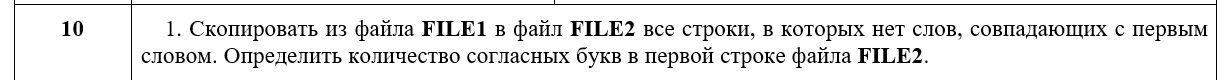
Минск, 2024

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 10



#include <iostream>

#include <fstream>

#include <string> // подключаем библиотеку для работы со строками

using namespace std; // подключаем пространство имён

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 где-то ещё в строке

{

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 бешать 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;

}

}

// так как предыдущее условие не сработало значит мы возможно нашли нужное слово

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 << firstWord<<endl; // выводим первое слово

// возвращаем позицию чтения файла на начало

file1.clear();

file1.seekg(0);

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

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

while (getline(file1, line)) // цикл для пробега по всем строкам из файла тоесть запись в одной итерции цикла одной строки 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

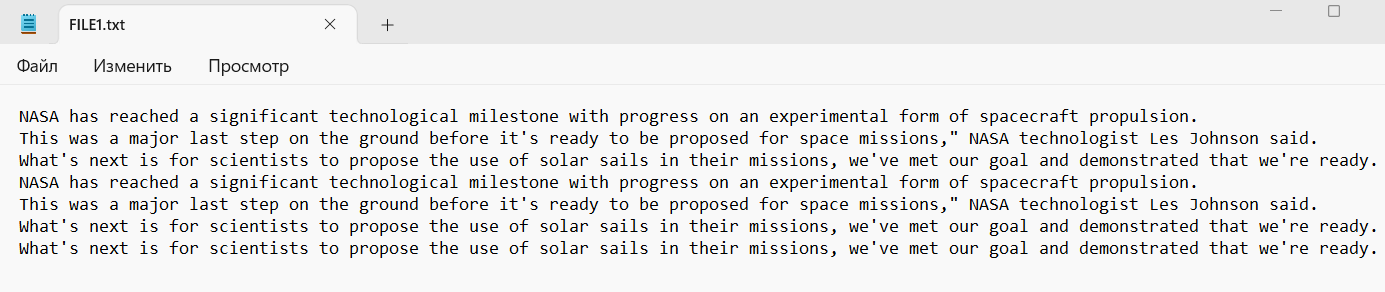
file1.close();

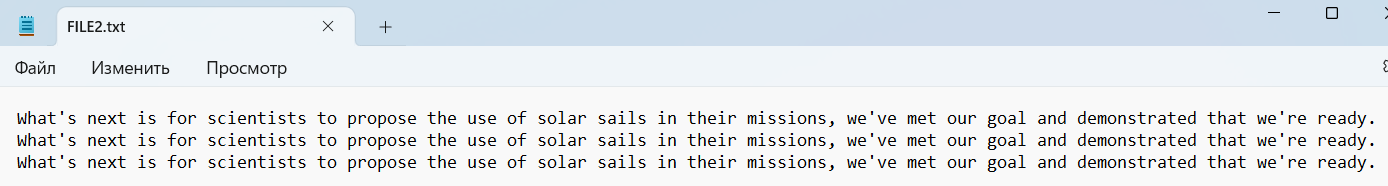
file2.close();

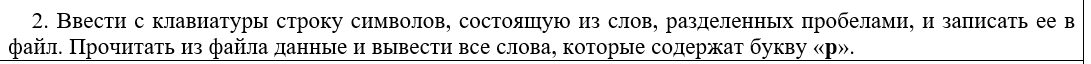
return 0;

}









#include <iostream>

#include <fstream> // для работы с файлами

#include <string> // для работы со строками

using namespace std;

bool containsP(const string& word) // функция для поиска P в слове

{

for(int i = 0; i < word.length(); i++) // бежим по слову

{

if (word[i]== 'p' || word[i] == 'P') //если есть P

{

return true; // то возвращаем true

}

}

return false;// если вслове нет P возвращаем false

}

int main()

{

setlocale(LC\_ALL, "rus");

string str;

cout << "Введите строку: ";

getline(cin, str); // вводим строку

cout << endl; // переход на новую строку в консоли

ofstream file("FILE.txt"); // открываем файл для записи

if (file.is\_open()) // проверка на открытие

{

file << str; // записываем строку в файл

file.close(); // закрываем файл

}

else // иначе выводим ошибку

{

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

return 1;

}

ifstream file1("FILE.txt"); // открываем файл для чтения

string word; // переменная для хранения текущего слова

if (file1.is\_open()) // если файл открыт

{

while (file1 >> word) //цикл который записывает в word слова по очереди, пока не дойдёт до конца файла

{

if (containsP(word)) // вызываем функцию и если условие выполняется

{

cout << word << " "; // выводим слово

}

}

file1.close(); // закрываем файл

}

else // иначе выводим ошибку

{

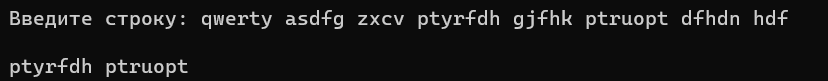
cout << "Не удалось открыть файл для чтения.";

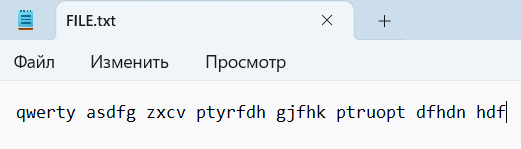
return 1;

}

return 0;

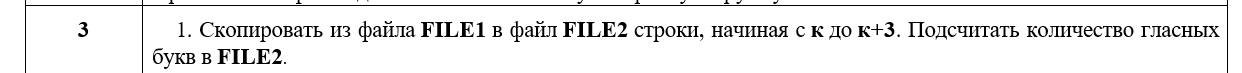
}





Доп. Задачи

Вариант 3



#include <iostream>

#include <fstream>

using namespace std;

bool isVow(char\* symbol) // функция для подсчёта гласных букв

{

char s = \*symbol;

if (s == 'a' || s == 'e' || s == 'i' || s == 'o' || s == 'u' || s == 'A' || s == 'E' || s == 'I' || s == 'O' || s == 'U') // если буква гласная

{

return true; // возвращаем true

}

else

{

return false; // иначе false

}

}

int main()

{

setlocale(LC\_ALL, "ru");

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

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

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;

}

int k;

char str[1000]; // переменная для хранения строки

cout << "Введите k от 1 до 6: "; cin >> k; // вводим K

if (k > 6 || k<1) // так как в файле 9 строк ставим такое условие

{

cout << "Вы ввели неправильное значение! ";

return 1;

}

int nk = k + 3; // переменная которая хранит по какую строку мы будем выводить

int count = 1; // счтётчик строк

while (file1.getline(str, 1000)) // цикл для записи строк из файла

{

if (count >= k && count<=nk) // если выполняется условие

{

file2 << str<<"\n"; // записываем строку в файл

}

count++; // увеличиваем знаение

}

// закрываем файлы

file1.close();

file2.close();

ifstream file3("FILE2.txt"); // открываем файл для чтения

if (file3.fail()) // проверка на открытие файла

{

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

return 1;

}

count = 0; // обнуляем счётчик для дальнейшего сипользования при подсчёте гласных букв

while (file3.getline(str, 1000)) // цикл для записи строк из файла

{

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

{

if (isVow(&str[i])) // вызваем функцию и если условие выполняется

{

count++; // то увеличиваем значение

}

}

}

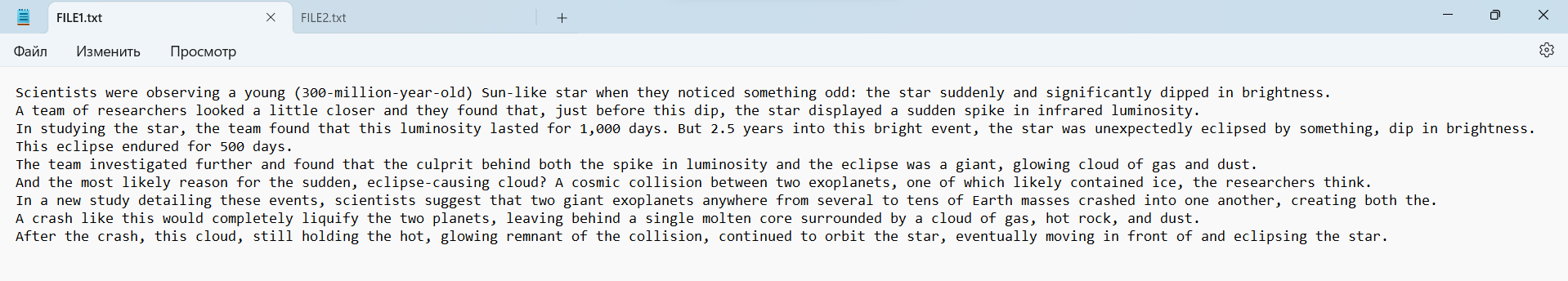
cout << "Количество гласных: " << count; // выводим количество в консоль

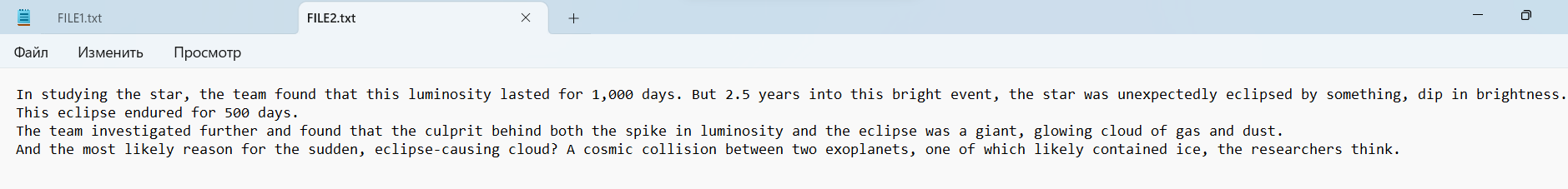
file3.close(); // закрываем файл

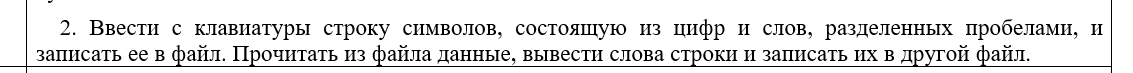
return 0;

}









#include <iostream>

#include <fstream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "ru");

// открываем 2 файла для записи

ofstream file1("FILE1.txt");

ofstream file2("FILE2.txt");

char str[1000]; // переменная для хранения строки

cout << "Введите строку: ";

gets\_s(str); // вводим строку

if (file1.is\_open()) // если файл открылся

{

file1 << str; // записываем в него строку

file1.close(); // закрываем файл

}

else // иначе выводим ошибку

{

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

return 1;

}

//открываем файл для чтения

ifstream file3("FILE1.txt");

if (file3.fail()) // проверка на открытие файла

{

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

return 1;

}

char word[100]; // переменная для хранения слова

while (file3 >> word) // цикл для записи слов из файла

{

int i = 0;

if (!(word[i]>='0' && word[i]<='9')) // если word не содержит цифру значит группа цифр

{

file2 << word << '\n'; // поэтому записываем в файл

cout << word << " "; // выводим в консоль

}

}

//закрываем файлы

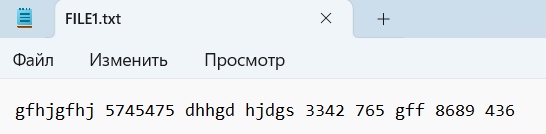
file2.close();

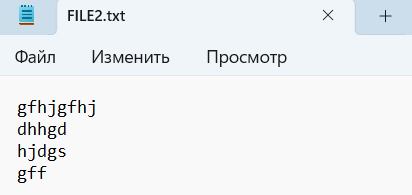
file3.close();

return 0;

}







Вариант 11





#include <iostream>

#include <fstream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

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

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

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;

}

char str[1000]; // переменная которая хранит строки

while (file1.getline(str, 1000)) // цикл для записи строк из файла

{

int count = 0; // переменная для хранения количества пробелов

for (int i = 0;i < strlen(str);i++) // цикл для пробега по всей строке

{

if (str[i] == ' ') // если нашли пробел

{

count++; // увеличваем значение

}

}

if (count == 0) // если пробелов нет, значит строка состоит из одного слова

{

file2 << str << '\n'; // записываем её в файл

}

}

// закрываем файлы

file1.close();

file2.close();

ifstream file3("FILE2.txt"); //открываем файл для чтения

if (file3.fail()) // проверка на открытие файла

{

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

return 1;

}

int count = 0; // переменная для подсчёта символов

while (file3.getline(str, 1000)) // цикл для записи строк

{

count += strlen(str); // считаем количество символов

}

cout << "Количество символов: " << count; // выводим в консоль

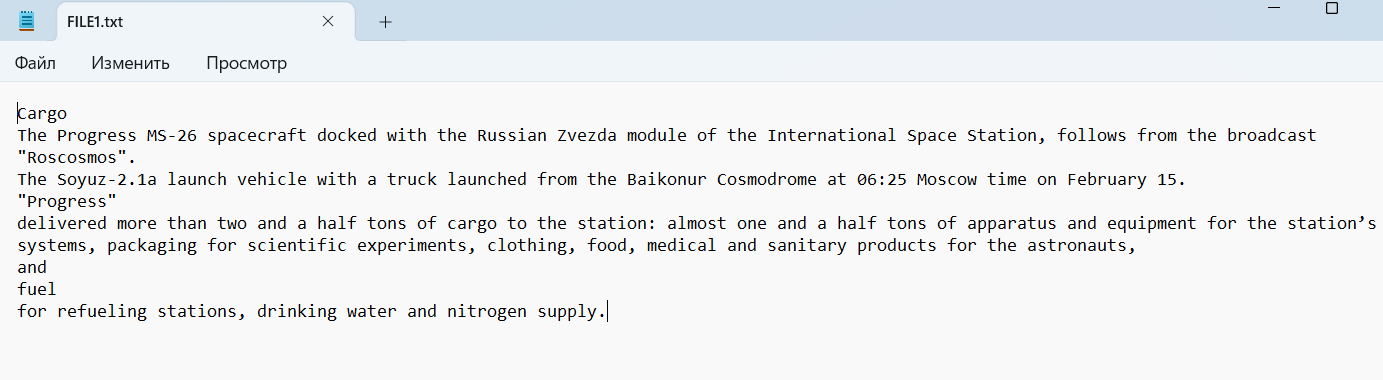
// закрываем файл

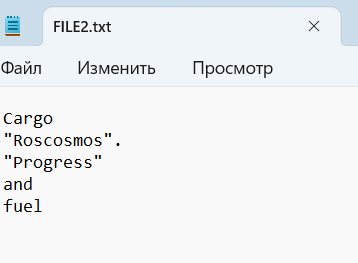
file3.close();

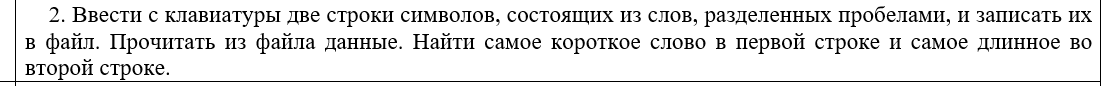
return 0;

}









#include <iostream>

#include <fstream>

using namespace std;

void findMin(char\* str) // функция для короткого слова

{

char word[100]; // переменная для хранения этого слова

int minlen = INT\_MAX; // переменная для хранения длины минимального слова

int len = 0; // переменная для хранения текущего слова

int m = 0; // переменная для хранения начала текущего слова

for (int i = 0;i < strlen(str);i++) // бежим по всей строке

{

if (str[i] == ' ' || str[i + 1] == '\0') // если текущая позиция пробел или нуль терминатор ,то

{

if (len < minlen) // если длина текущего слова меньше минимального

{

minlen = len; // присваиваем это значение

for (int j = 0; j < len; j++) // записываем это слово в переменную

{

word[j] = str[m + j];

}

word[len] = '\0'; // добавляем нуль терминатор в конец

}

len = 0; // обнуляем

m = i + 1; // увеличиваем знаение чтобы не начиналось с пробела

}

else

{

len++; // увеличиваем значение

}

}

cout << "Наименьшее слово в первой строке: " << word << endl; // выводим это слово

}

void findMax(char\* str) // функция для поиска длинного слова

{

char word[100]; // переменная для хранения слова

int maxlen = INT\_MIN, len=0, m=0; // тоже самое что и впрошлой функции

for (int i = 0;i < strlen(str);i++) // бежим по всей строке

{

if (str[i] == ' ' || str[i + 1] == '\0') // если текущая позиция пробел или нуль терминатор ,то

{

if (len > maxlen) // если длина текущего слова больше максимального

{

maxlen = len; // присваиваем это значение

for (int j = 0; j < len; j++) // записываем это слово в переменную

{

word[j] = str[m + j];

}

word[len] = '\0'; // добавляем нуль терминатор в конец

}

len = 0; // обнуляем

m= i + 1; // увеличиваем знаение чтобы не начиналось с пробела

}

else

{

len++; // увеличиваем значение

}

}

cout << "Набольшее слово во второй строке: " << word << endl; // выводим это слово

}

int main()

{

setlocale(LC\_ALL, "rus");

char line1[500], line2[500];

cout << "Введите первую строку: "; gets\_s(line1); // записываем первую строку

cout << endl << "Введите вторую строку: "; gets\_s(line2); // записываем вторую строку

ofstream file1("FILE1.txt"); // открываем файл для записи

if (file1.is\_open()) // если файл отркылся

{

file1 << line1 <<'\n' << line2; // записываем строки

file1.close(); // закрываем файл

}

else // иначе выводим ошибку

{

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

return 1;

}

ifstream file2("FILE1.txt"); // открываем файл для чтения

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

{

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

return 1;

}

char str[1000]; // переменная для хранения строки

file2.getline(str, 1000); // записываем первую строку

findMin(str); // вызваем функцию для поиска нужного слова

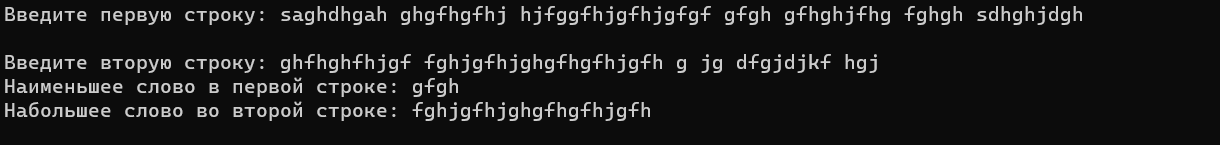
file2.getline(str, 1000); // записываем вторую строку

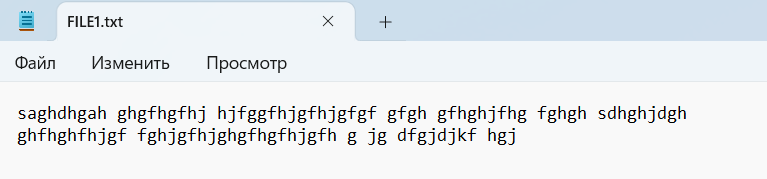
findMax(str); // вызваем функцию для поиска нужного слова

file2.close(); // закрываем файл

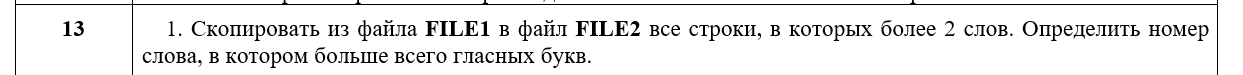
return 0;

}





Вариант 13



#include <iostream>

#include <fstream>

using namespace std;

bool isVow(char\* symbol) //функция для проверки на гласную

{

char s = \*symbol;

if (s == 'a' || s == 'e' || s == 'i' || s == 'o' || s == 'u' || s == 'A' || s == 'E' || s == 'I' || s == 'O' || s == 'U') // если символ гласный

{

return true; // возвращаем true

}

else

{

return false; // иначе возвращаем false

}

}

int findNumb(char\* str) // функция которая определяет номер нужного слова

{

int count = 0; // переменная для подсчёта количества гласных в нужном слове

int nowcount = 0; // переменная для подсчёта количества гласных в текущем слове

int bufnumbword = 0; // переменная для подсчёта всех слов

int numbword = 0; // переменная для хранения номера нужного слова

for (int i = 0;i < strlen(str);i++) // бежим по строке

{

if (isVow(&str[i])) // если гласная

{

nowcount++; // увеличиваем значение

}

if (str[i] == ' ' || str[i + 1] == '\0') // если слово закончилось

{

bufnumbword++; //увеличваем значение

if (count < nowcount) // если в текущем слове гласных больше, то

{

//обновляем значения

count = nowcount;

numbword = bufnumbword;

}

nowcount = 0; // обнуляем значение

}

}

return numbword; // возвращаем номер слова

}

int main()

{

setlocale(LC\_ALL, "rus");

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

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

char line[1000]; // переменная для хранения строки

int count = 0; // счётчик пробелов

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

{

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

return 1;

}

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

{

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

return 1;

}

while (file1.getline(line, 1000)) // цикл для записи строк из фала

{

for (int i = 0; i < strlen(line);i++) // бежим по строке

{

if (line[i] == ' ') // если наткунлись на пробел

{

count++; // увеличиваем значение

}

}

if (count > 1) // если пробелов больще 1, значит слов больше чем два

{

file2 << line<< '\n'; // поэтому записываем в файл

int numb = findNumb(line); // вызываем функцию для поиска номера слова и присваиваем её значение numb

cout << "Номер слова: " << numb << endl; // выводим номер слова

}

count = 0; // обнуляем счётчик

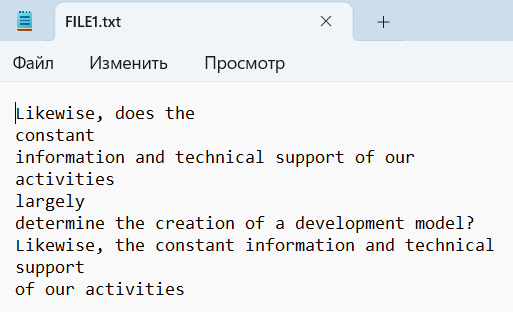
}

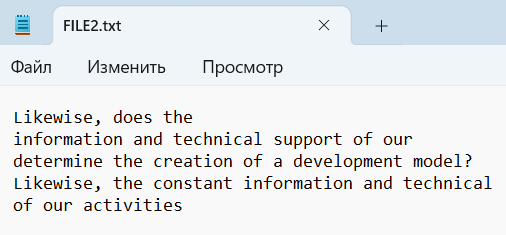
// закрываем файлы

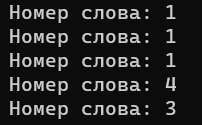
file1.close();

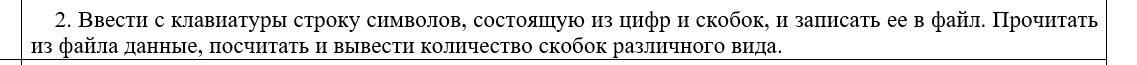
file2.close();

}









#include <iostream>

#include <fstream>

using namespace std;

bool isBracket(char\* str) // функция для поиска скобок

{

char c = \*str;

if ( c == '('|| c == ')' || c=='['|| c == ']' || c == '{' || c == '}') // если текущий символ скобка

{

return true; // возвращаем true

}

else

{

return false; // иначе возвращаем false

}

}

int main()

{

setlocale(LC\_ALL, "rus");

ofstream file1("FILE1.txt"); // открываем файл для записи

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

{

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

return 1;

}

char str[1000]; // переменная для хранения строки

cout << "Введите строку: ";

gets\_s(str); // вводим строку

file1 << str; // записываем в файл

file1.close(); // закрываем файл

ifstream file2("FILE1.txt"); // открываем файл для чтения

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

{

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

return 1;

}

char newstr[1000]; // переменная для хранения новой строки

file2.getline(newstr, 1000); // записываем строку из файла

int count=0; // счётчик скобок

for (int i = 0; i < strlen(newstr);i++) // бежим по строке

{

if (isBracket(&newstr[i])) // если скобка

{

count++; // увеличиваем значение

}

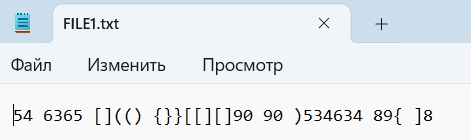
}

cout << "Количество скобок: " << count; // выводим значение

file2.close(); // закрываем файл

}

****

****