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:

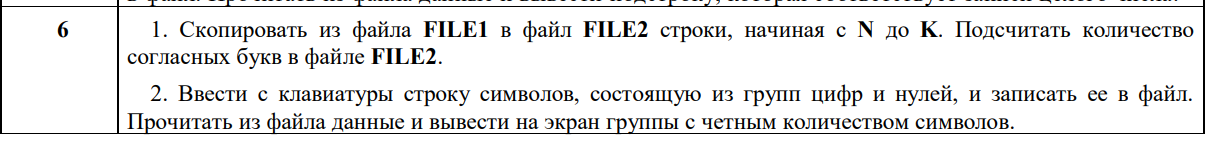
Студент 1 курса 8 группы

Лужецкий Владислав Константинович

Преподаватель: асс. Андронова М.В.

2024, Минск

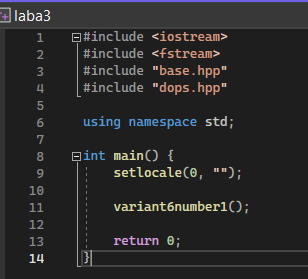
**Основа:**

****

**№1**

//функция вызывается в main. Сам код прописал в заголовочных файлах base.hpp и dops.hpp





using namespace std;

int variant6number1()

{

int N, K; // Переменные для хранения пользовательского ввода

int counter = 0; // Счетчик согласных букв

char buffer[255]; // Буфер для чтения строк из файла

ifstream inputFile("FILE1.txt"); // Открытие файла для чтения

if (!inputFile.is\_open()) {

cout << "Не удалось открыть файл A.txt" << endl; // Вывод сообщения об ошибке, если файл не удалось открыть

return 0;

}

cout << "Введите N: "<< endl;

cin >> N;

cout << "Введите k: "<< endl;

cin >> K;

int i=0;

ofstream outputFile("FILE2.txt"); // Открытие файла для записи

while (inputFile.getline(buffer, 255))

{

if (i >= N && i <= K)

{

outputFile << buffer << "\n"; // Запись строки в файл

for (int j = 0; j < strlen(buffer); j++)

{

if (buffer[j] != 'a' && buffer[j] != ' ' && buffer[j] != 'A' && buffer[j] != 'e' && buffer[j] != 'E' && buffer[j] != 'i' && buffer[j] != 'I' && buffer[j] != 'o' && buffer[j] != 'O' && buffer[j] != 'u' && buffer[j] != 'U')

{

counter++; // Увеличение счетчика, если символ является согласной буквой

}

}

}

i++;

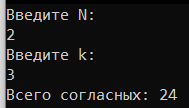
}

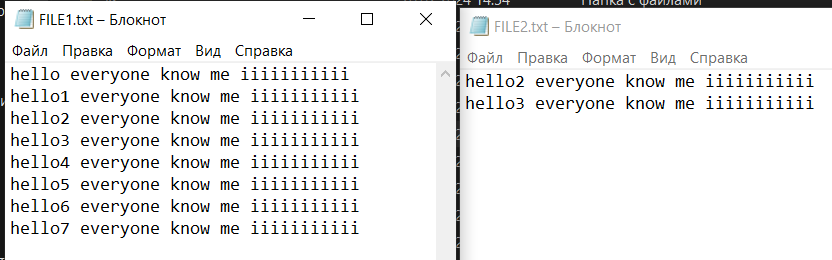
inputFile.close(); // Закрытие файла

outputFile.close(); // Закрытие файла

cout << "Всего согласных: " << counter; // Вывод количества согласных букв

return 0;

}****

****

**№2**

int variant6number2()

{

// Ввод строки с клавиатуры

cout << "Введите строку символов, состоящую из групп цифр и нулей: ";

string inputString;

getline(cin, inputString);

// Запись строки в файл

ofstream outputFile("input.txt");

if (!outputFile.is\_open()) {

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

return 1;

}

outputFile << inputString;

outputFile.close();

// Чтение данных из файла и вывод групп с четным количеством символов

ifstream inputFile("input.txt");

if (!inputFile.is\_open()) {

cerr << "Не удалось открыть файл для чтения." << endl;

return 1;

}

string line;

while (inputFile >> line) {

if (line.size() % 2 == 0) {

cout << line << endl;

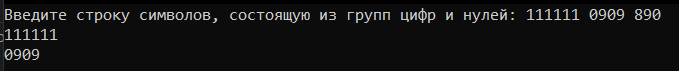
}

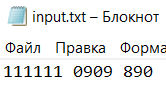
}

inputFile.close();

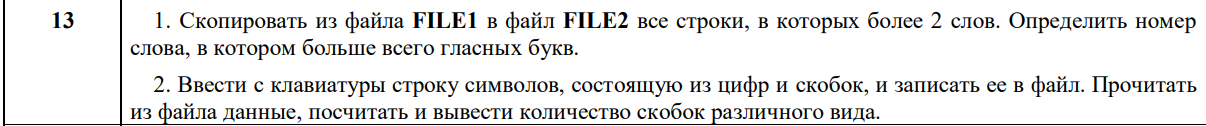
return 0;

}





**Допы:**

****

**№1**

#include <string>

#include <sstream>

using namespace std;

// Функция для подсчета гласных в слове

int countVowels(const string& word) {

int count = 0;

string vowels = "aeiouAEIOU";

for (char c : word) {

if (vowels.find(c) != string::npos) {

count++;

}

}

return count;

}

// Функция для определения номера слова с наибольшим количеством гласных

int findMaxVowelWord(const string& line) {

istringstream iss(line);

string word;

int maxVowelCount = 0;

int maxVowelWordIndex = -1;

int currentIndex = 0;

while (iss >> word) {

int vowelCount = countVowels(word);

if (vowelCount > maxVowelCount) {

maxVowelCount = vowelCount;

maxVowelWordIndex = currentIndex + 1; // С учетом 1-индексации

}

currentIndex++;

}

return maxVowelWordIndex;

}

int variant13number1() {

ifstream inputFile("FILE1.txt"); // Открытие файла для чтения

ofstream outputFile("FILE2.txt"); // Открытие файла для записи

if (!inputFile.is\_open() || !outputFile.is\_open()) {

cerr << "Не удалось открыть файлы." << endl; // Вывод сообщения об ошибке, если не удалось открыть файлы

return 1;

}

std::string line;

while (getline(inputFile, line)) {

istringstream iss(line);

string word;

int wordCount = 0;

while (iss >> word) {

wordCount++;

}

if (wordCount > 2) {

int maxVowelWordIndex = findMaxVowelWord(line);

outputFile << line << " (Слово с наибольшим количеством гласных - " << maxVowelWordIndex << ")\n"; // Запись строки с информацией о слове с наибольшим количеством гласных

}

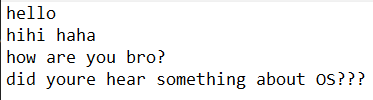
}

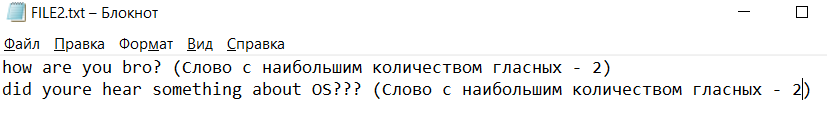
inputFile.close(); // Закрытие файла

outputFile.close(); // Закрытие файла

return 0;

}

****

****

**№2**

#include <unordered\_map>

using namespace std;

int variant13number2() {

setlocale(LC\_ALL, "rus");

// Ввод строки с клавиатуры

cout << "Введите строку символов, состоящую из цифр и скобок: ";

string inputString;

getline(cin, inputString);

// Запись строки в файл

ofstream outputFile("input.txt");

if (!outputFile.is\_open()) {

cerr << "Не удалось открыть файл для записи." << endl; // Вывод сообщения об ошибке, если не удалось открыть файл для записи

return 1;

}

outputFile << inputString;

outputFile.close();

// Чтение данных из файла и подсчет количества различных скобок

ifstream inputFile("input.txt");

if (!inputFile.is\_open()) {

cerr << "Не удалось открыть файл для чтения." << endl; // Вывод сообщения об ошибке, если не удалось открыть файл для чтения

return 1;

}

unordered\_map<char, int> bracketCounts;

char c;

while (inputFile.get(c)) {

if (c == '(' || c == ')' || c == '[' || c == ']' || c == '{' || c == '}') {

bracketCounts[c]++;

}

}

inputFile.close();

// Вывод количества скобок различного вида

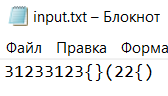
cout << "Количество различных скобок:\n";

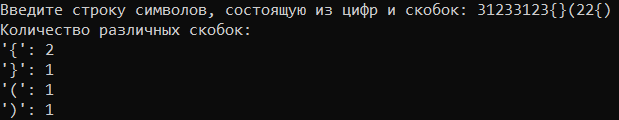
for (const auto& pair : bracketCounts) {

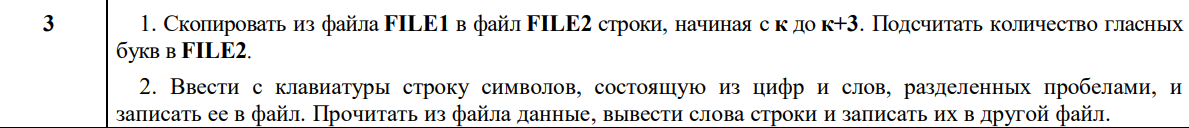
cout << "'" << pair.first << "': " << pair.second << endl;

}

return 0;

}****





№1

bool isVowel(char\* symb) {

char s = \*symb;

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

return true;

else

return false;

}

// Функция для извлечения строк из одного файла, записи их в другой файл и подсчета количества гласных букв

int extractLinesAndCountVowels() {

ifstream file1("FILE1.txt"); // Открытие файла для чтения

ofstream file2("FILE2.txt"); // Открытие файла для записи

int k;

char str[1000];

cout << "Введите k от 1 до 7: ";

cin >> k;

if (k > 7) {

cout << "Вы ввели неправильное значение! ";

return 1;

}

int k2 = k + 3;

int count = 1;

while (file1.getline(str, 1000)) { // Чтение строк из файла file1

if (count >= k && count <= k2) { // Проверка, находится ли текущая строка в заданном диапазоне k-k2

file2 << str << "\n"; // Запись строки в файл file2

}

count++;

}

file1.close(); // Закрытие файла file1

file2.close(); // Закрытие файла file2

ifstream file3("FILE2.txt"); // Открытие файла для чтения

count = 0;

while (file3.getline(str, 1000)) { // Чтение строк из файла file3

for (int i = 0; str[i] != '\0'; i++) { // Перебор символов в строке

if (isVowel(&str[i])) { // Проверка, является ли текущий символ гласной буквой

count++; // Увеличение счетчика гласных букв

}

}

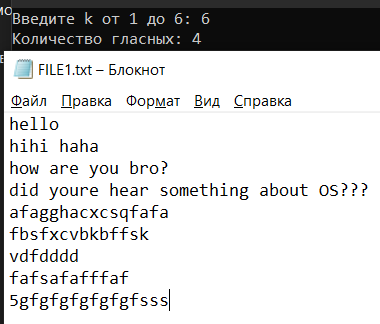
}

file3.close(); // Закрытие файла file3

cout << "Количество гласных: " << count;

return 0;

}



№2

int variant3number2() {

setlocale(LC\_ALL, "ru");

ofstream file1("FILE1.txt"); // Открытие файла для записи

ofstream file2("FILE2.txt"); // Открытие файла для записи

cout << "Введите строку: ";

char str[1000];

cin.getline(str, 1000); // Ввод строки с пробелами

if (file1.is\_open()) {

file1 << str; // Запись строки в файл file1

file1.close(); // Закрытие файла file1

} else {

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

return 1;

}

ifstream file3("FILE1.txt"); // Открытие файла для чтения

char word[100];

while (file3 >> word) { // Чтение слов из файла file3

int i = 0;

// Проверка, является ли первый символ слова цифрой

if (!(word[i] >= '0' && word[i] <= '10')) {

file2 << word << '\n'; // Запись слова в файл file2

cout << word << " "; // Вывод слова на экран

}

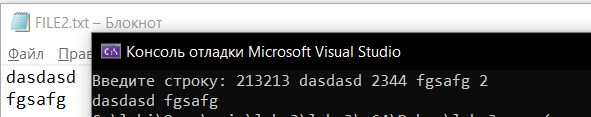
}

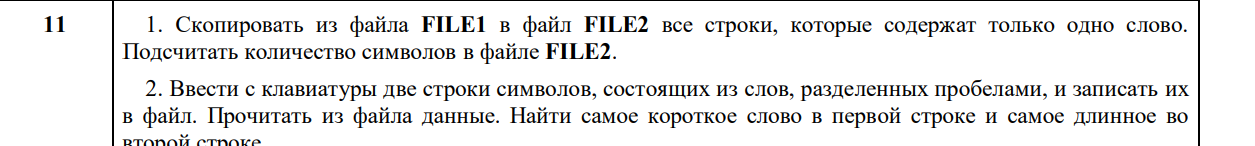
file2.close(); // Закрытие файла file2

file3.close(); // Закрытие файла file3

return 0;

}





№1

int variant11number1() {

ifstream file1("FILE1.txt"); // Открытие файла для чтения

ofstream file2("FILE2.txt"); // Открытие файла для записи

char str[1000];

while (file1.getline(str, 1000)) { // Чтение строк из файла file1

int count = 0;

for (int i = 0; i < strlen(str); i++) { // Подсчет пробелов в текущей строке

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

count++;

}

}

if (count == 0) { // Если в строке нет пробелов, записываем ее в файл file2

file2 << str << '\n';

}

}

file1.close(); // Закрытие файла file1

file2.close(); // Закрытие файла file2

ifstream file3("FILE2.txt"); // Открытие файла для чтения

int count = 0;

while (file3.getline(str, 1000)) { // Чтение строк из файла file3

for (int i = 0; i < strlen(str); i++) { // Подсчет символов в текущей строке

if (str[i] != '\0') {

count++;

}

}

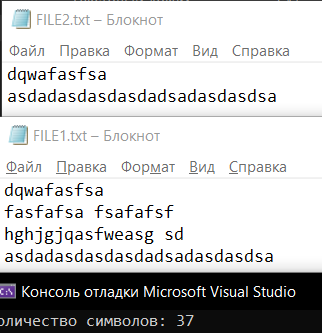
}

cout << "Количество символов: " << count;

file3.close(); // Закрытие файла file3

return 0;

}



№2

int variant11number2() {

setlocale(LC\_ALL, "rus");

char str1[500], str2[500];

cout << "Введите первую строку: ";

gets\_s(str1); // Ввод первой строки

cout << endl << "Введите вторую строку: ";

gets\_s(str2); // Ввод второй строки

ofstream file1("FILE1.txt");

if (file1.is\_open()) {

file1 << str1 << '\n' << str2; // Записываем строки в файл

file1.close();

}

else {

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

return 1;

}

ifstream file2("FILE1.txt");

char str[1000];

file2.getline(str, 500); // Читаем первую строку из файла

int size = strlen(str);

char word[100];

int minlen = INT\_MAX, buflen = 0, k = 0;

// Поиск наименьшего слова в первой строке

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

if (str[i] == ' ' || str[i + 1] == '\0') {

if (buflen < minlen) {

minlen = buflen;

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

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

}

word[buflen] = '\0';

}

buflen = 0;

k = i + 1;

}

else {

buflen++;

}

}

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

file2.getline(str, 1000); // Читаем вторую строку из файла

k = 0, buflen = 0;

int maxlen = INT\_MIN;

// Поиск наибольшего слова во второй строке

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

if (str[i] == ' ' || str[i + 1] == '\0') {

if (buflen > maxlen) {

maxlen = buflen;

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

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

}

word[buflen] = '\0';

}

buflen = 0;

k = i + 1;

}

else {

buflen++;

}

}

cout << "Наибольшее слово во второй строке: " << word << endl;

file2.close();

return 0;

}

