Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(ПНИПУ)

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №9

по теме:

СТРОКОВЫЙ ВВОД-ВЫВОД

Выполнила: студентка группы РИС-22-1б

Черкасова А.А\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверил: доцент кафедры ИТАС

Полякова О.А\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_/\_\_\_\_\_\_\_/\_\_\_\_\_г.

Пермь 2023

**Содержание**

[**Введение** 3](#_Toc127379672)

[**Анализ задачи** 3](#_Toc127379673)

[**Блок – схема** 4](#_Toc127379674)

[**Приложение А** 7](#_Toc127379675)

[**Приложение Б** 9](#_Toc127379676)

# **Введение**

**Постановка задачи:**

А) Скопировать из файла F1 в файл F2 все строки, в которых нет одинаковых слов.

Б) Определить количество гласных букв в первой строке файла F2.

**Цель:** Работа с текстовыми файлами, ввод-вывод текстовой информации и ее хранение на внешних носителях.

Для достижения поставленной цели, необходимо решить следующие **задачи**:

* Провести анализ задачи
* Реализовать задачу на языке С++
* Составить блок-схему

# **Анализ задачи**

1. В файле F1 изначально присутствует текст, который требуется обработать.

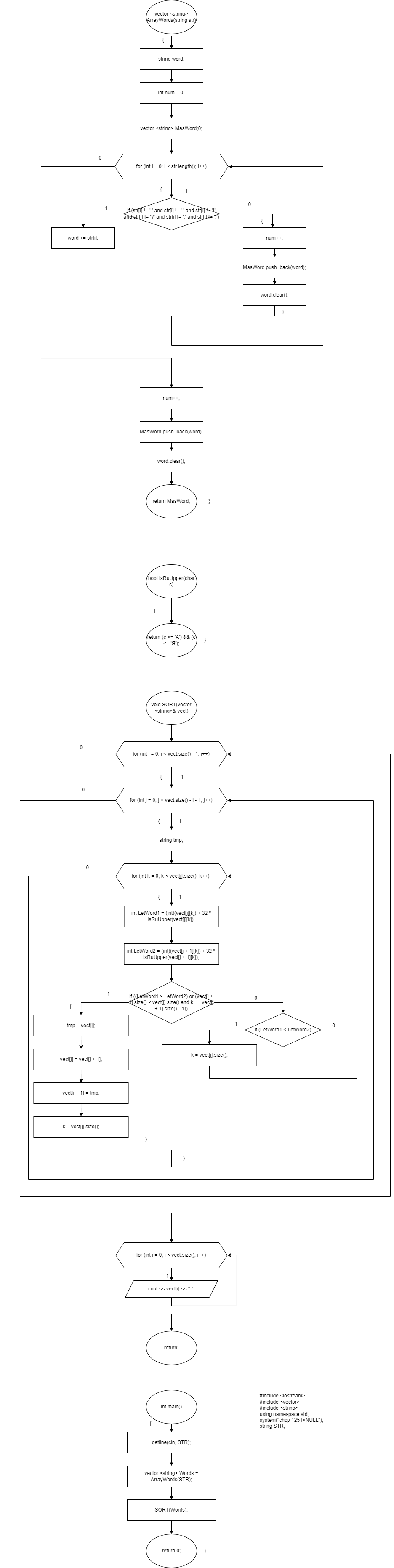
2. Файл F1 открывается в режиме чтения*.* Сначала цикл работает до тех пор, пока не дойдет до конца файла и считает количество строк в файле. После того, как стало известно количество строк, создаётся массив этих строк и возвращается функцией.

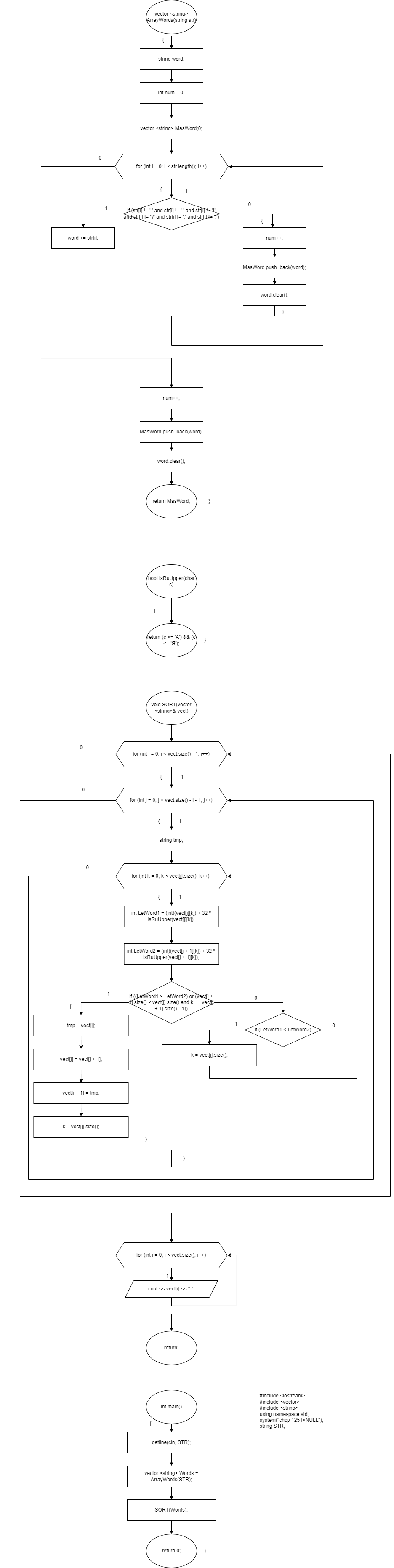
3. В другой функции разделяются строки на слова и далее идет сверка слов сначала по длине, а далее побуквенно. Если нашли слово в строке, то номер строки помещается в вектор, тем самым получается вектор, хранящий номера строк, имеющих одинаковые слова.

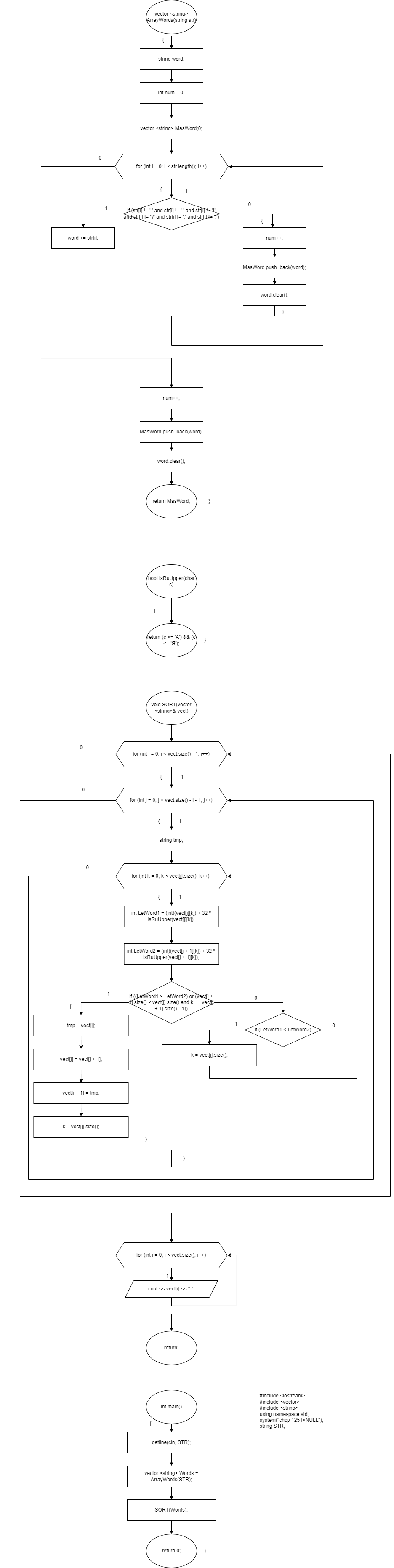
4. Далее, открывается файл F2 для записи. С помощью цикла проходимся по массиву строк, и если индекс строки файла F1 не совпадает с индексом, хранящемся в векторе, то строка записывается в файл F2.

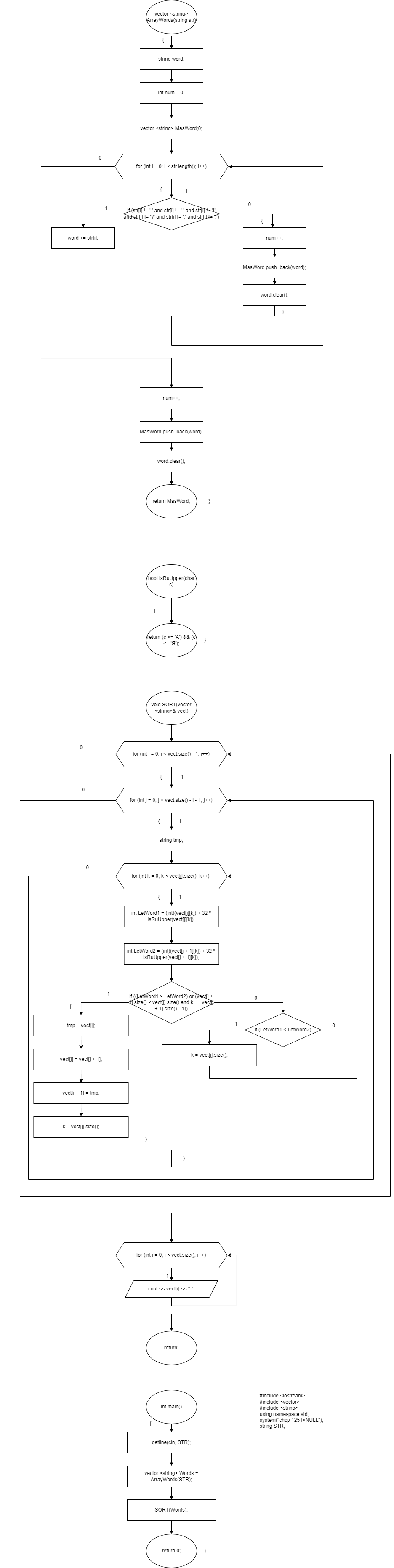
5. Подсчёт количества гласных. Создаётся строка, содержащая английские и русские гласные буквы, через цикл проходимся по словам первой строки файла F2. Буквы слов сравниваются с буквами строки «алфавита», если нашлось совпадение, то к счётчику гласных букв прибавляется единица.

## **Блок – схема**

**

**

**

**

# **Приложение А**

# **Листинг программы**

/\*

1) Скопировать из файла F1 в файл F2 все строки, в которых нет одинаковых слов.

2) Определить количество гласных букв в первой строке файла F2.

\*/

#include <iostream>

#include <string>

#include <fstream>

#include <vector>

#include <algorithm> // библиотека дляприменения алгоритма поиска в векторе

using namespace std;

string fi1 = "F1.txt"; // файл для чтения

string fi2 = "F2.txt"; //файл для записи

ifstream fin; // чтение из файла F1

ofstream fout; // запись в файл

ifstream buf; // чтение из файла F2

int N; // количество строк

string\* ReadFile()

{

fin.open(fi1);

if (!fin.is\_open())

{

cout << "Ошибка открытия файла F1 для чтения!" << endl;

}

else

{

cout << "Файл F1 для чтения открыт!" << endl;

string str; // для работы get

while (!fin.eof()) // для подсчета количества строк

{

getline(fin, str); // считывание строки

N++;

}

cout << N << " строки в файле." << endl;

fin.seekg(0); //курсор в начало файла

string\* stroks = new string[N]; //массив строк

int i = 0;

while (!fin.eof()) // для подсчета количества строк

{

getline(fin, stroks[i]); // считывание строки

i++;

}

return stroks;

}

fin.close();

}

bool IsRuUpper(char c) // проверка русской буквы на регистр

{

return (c >= 'А') && (c <= 'Я');

}

int SIZE = 0; //количество строк с повторяющимися словами

vector <int> Search(string\* temp)

{

vector <int> NumStr; // вектор, хранящий номера строк, в которых есть повторяющиеся слова

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

{

vector <string> MasWord;

string word; // "временное" слово

int num = 0; // длина вектора слов

for (int j = 0; j < temp[i].length(); j++) // проход по всей длине строки

{

if (temp[i][j] != ' ' and temp[i][j] != '.' and temp[i][j] != ',' and temp[i][j] != '!' and temp[i][j] != '?' and temp[i][j] != ':' and temp[i][j] != ';')// "вычленение" слова из строки без учета знаков

word += temp[i][j]; // составление слова побуквенно

else

{

num++;

MasWord.push\_back(word);

word.clear();

}

}

num++;

MasWord.push\_back(word); // добавление слова в конец массива

word.clear();

bool fl = false; // флаг, обознач. наличие повтор. слов

int l = 0;

while (l < num and fl == false)

{

string TempWord = MasWord[l];

for (int j = l + 1; j < num; j++)

{

string Word = MasWord[j];

if (TempWord.length() == Word.length())

{

//bool fl2 = false;

for (int k = 0; k < Word.length(); k++)

{

int LetWord1 = (int)(TempWord[k]) + 32 \* IsRuUpper(TempWord[k]); // преобразование к одному регистру букв слов

int LetWord2 = (int)(Word[k]) + 32 \* IsRuUpper(Word[k]);

if (LetWord1 != LetWord2) //если буквы не совпадают, смысла проверять равенство слов

k = Word.length() + 1;

else

{

if (k == Word.length() - 1) //если дошли до конца слова и не вышли до этого цикла, значит слова одинаковые

{

fl = true; // это означает, что слова найдены

SIZE++; // увелииваем количество строк, с повтор. словами

NumStr.push\_back(i); // добавляем индекс строки, в которой есть повторени

}

}

}

}

}

l++;

}

}

return NumStr;

}

bool Check(vector <int> vect, int i) // Функция проверяет, есть ли индекс i в векторе индексов

{

bool fl = find(vect.begin(), vect.end(), i) != vect.end(); // https://www.techiedelight.com/ru/check-if-an-item-is-present-in-vector-in-cpp/

return fl;

}

void CopyFile(string\* array, vector <int> vect) // массив строк и вектор, содержащий номера строк, которые нам не подходят

{

fout.open(fi2);

if (!fout.is\_open())

{

cout << "Ошибка открытия файла F2 для записи!" << endl;

}

else

{

cout << "Файл F2 для записи открыт!" << endl;

string\* mas = new string[N - SIZE]; // массив, который не содержит строки с повор. словами

int index = 0;

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

{

if (!Check(vect, i))

{

mas[index] = array[i];

index++;

}

}

for (int i = 0; i < N - SIZE; i++) // заполнение файла данными

{

fout << mas[i] << endl;

}

cout << "Данные записаны в файл F2!" << endl;

}

fout.close();

}

void Counter()

{

buf.open(fi2);

if (!buf.is\_open())

{

cout << "Ошибка открытия файла F2 для чтения!" << endl;

}

else

{

cout << "Файл F2 для чтения открыт!" << endl;

string str;

getline(buf, str); // для работы get

int n = 0; // Счётчик кол-ва согласных букв

char symbol[] = "AaEeIiOoUuYyАаЕеЁёИиОоУуЫыЭэЮюЯя"; // Массив согласных

for (int i = 0; i < str.length(); i++) // Проход по всей длине строки

{

for (int j = 0; j < sizeof(symbol) / sizeof(symbol[0]); j++) // Проверка, символ - согласная буква?

{

if (str[i] == symbol[j]) // Если символ является согласной буквой, производится подсчёт символа

{

n++;

}

}

}

cout << endl;

buf.close();

cout << "Количество гласных букв в первой строке файла F2: " << n << endl;

}

}

int main()

{

system("chcp 1251>NULL");

string\* temp = new string[N];

temp = ReadFile();

for (int i = 0; i < N; i++) cout << temp[i] << endl;

vector <int> temp2 = Search(temp);

if (SIZE == 0) cout << "Нет строк с повторяющимися словами" << endl;

else

{

for (int i = 0; i < SIZE; i++) cout << temp2[i] << " ";

cout << endl;

CopyFile(temp,temp2);

Counter();

}

delete[] temp;

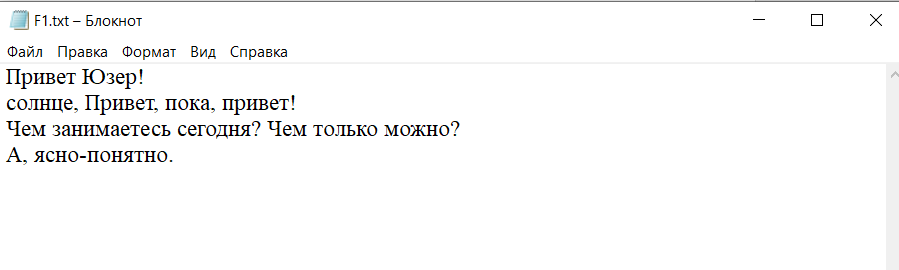
temp = nullptr;

}

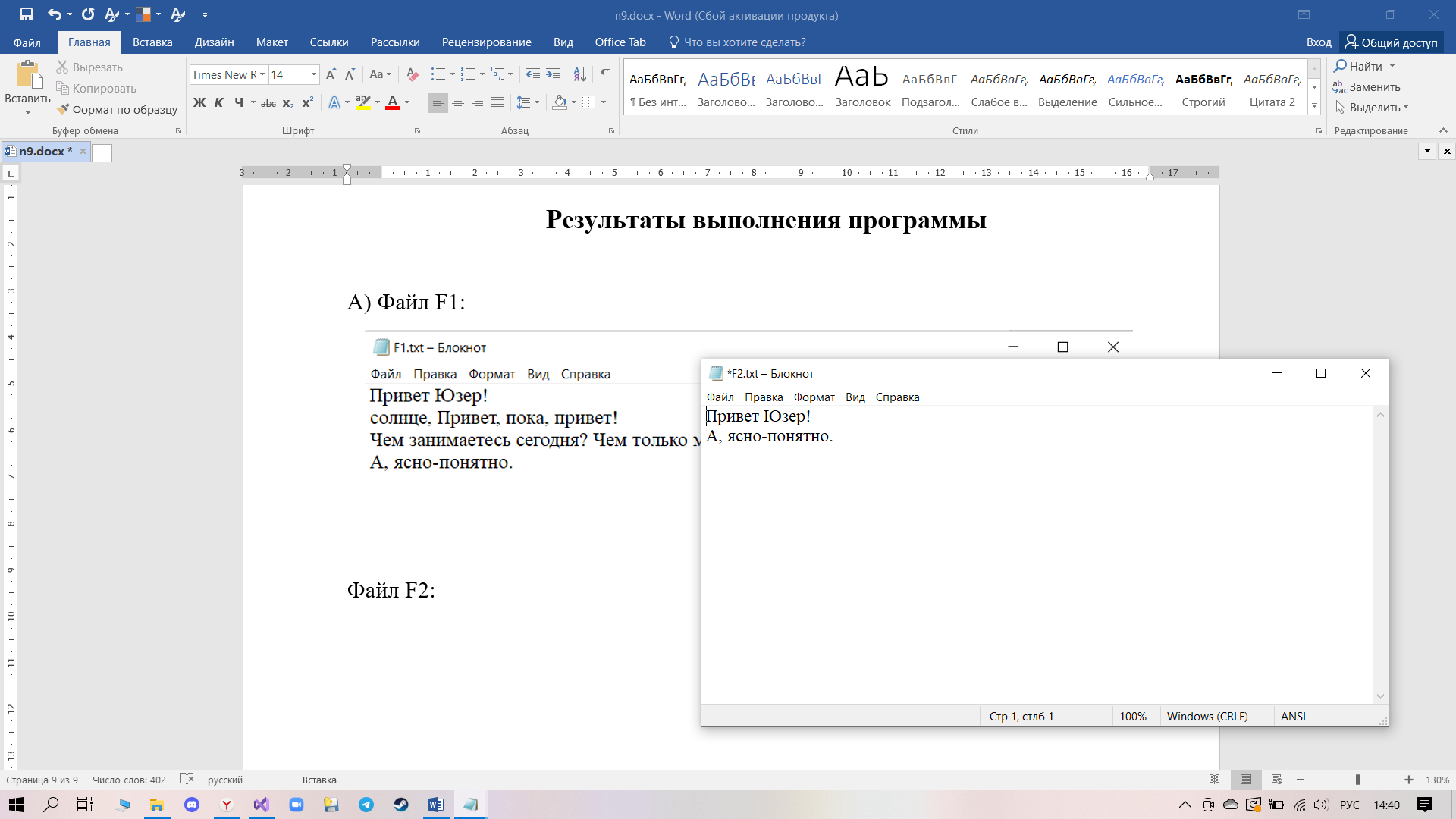
# **Приложение Б**

# **Результаты выполнения программы**

A) Файл F1:



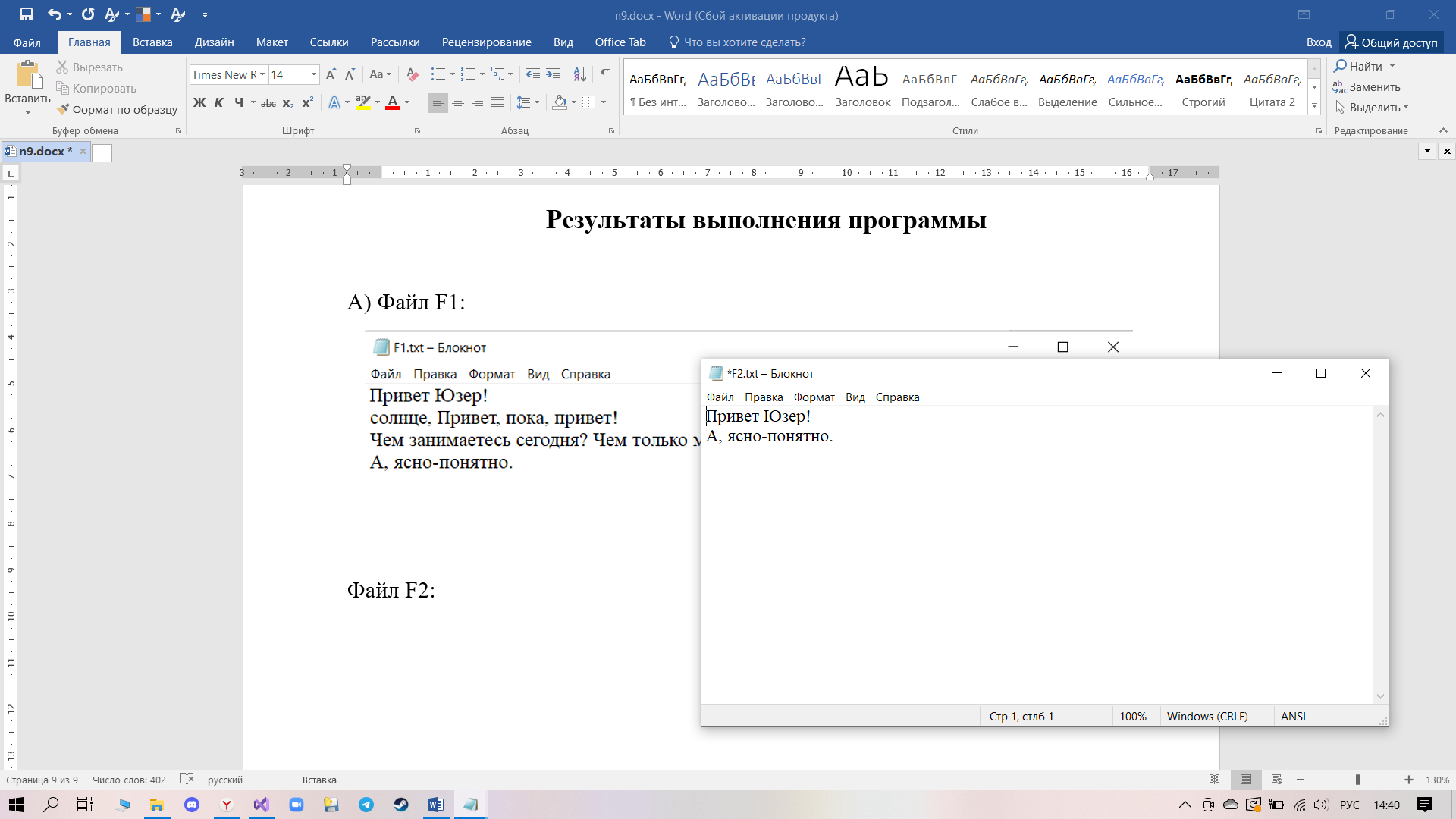
Файл F2:



Консоль:



Б) Файл F2:



Консоль:

