**«Пермский национальный исследовательский  
политехнический университет»**

**О Т Ч Ё Т**

**по лабораторной работе №9**

Тема: Структурное программирование. Файловые потоки

Вариант 5

Выполнил работу

студент группы ИВТ-20-2б

Нефедов Л.В.

Проверила

Доцент кафедры ИТАС

Полякова О.А.

Пермь, 2021

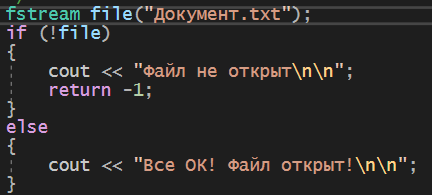
Постановка задачи

Создать текстовый файл F1 не менее, чем из 10 строк и записать в него информацию. После этого выполнить задание, а именно:

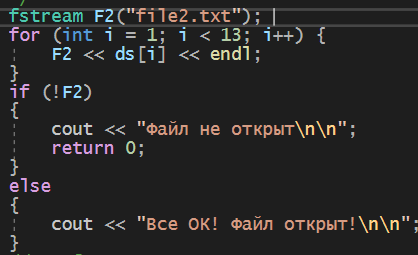
1. Скопировать из файла F1 в файл F2 строки, начиная с К до К+5.
2. Подсчитать количество гласных букв в файле F2.

Анализ задачи

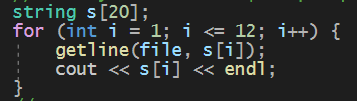
1. Для решения проблем чтения внешних файлов нужно:
   1. Проверить, существует ли файл ввода, откуда нужно достать информацию и в случае его отсутствия сообщить об ошибке.

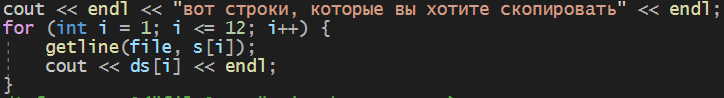


* 1. Затем, после совершения всех необходимых операций необходимо создать второй файл, куда будут записываться строки из первого файла, удовлетворяющие условию. Также надо проверить его на доступность (т.е. открыт ли он, прошла ли предыдущая операция создания корректно)

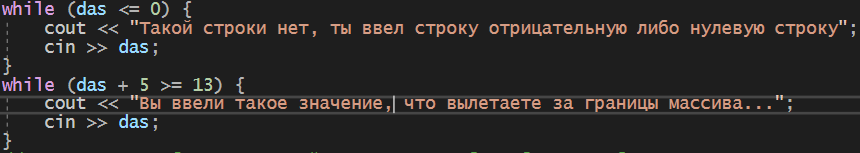


1. Для вывода в консоль строк из первого и второго файла нужно:
   1. Использовать стандартные операции вывода в циклах for с предварительной записью содержимого файлов в массивы типа string.

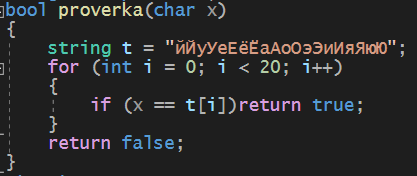




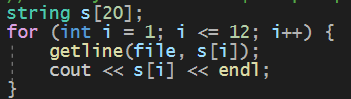
1. Для предотвращения ввода неверных данных надо:
   1. Использовать два цикла while с выводом конкретной ошибки. Поскольку пользователь может ввести неверную строку, с которой нужно начинать копирование, то надо использовать цикл while, чтобы в случае такой ошибки выводилась сама ошибка и пользователя просили бы ввести данные еще раз.



1. В ходе работы были использованы следующие типы данных:
   1. Несколько локальных и глобальных переменных типа int, которые являлись счетчиками циклов и переменной для подсчета количества гласных;
   2. Массивы типа string для записи строк и работы с ними;
   3. Для работы файлов—их чтения и записи были использованы классы fstream.
2. Также в ходе работы была разработана и применена функция proverka, которая ищет количество гласных букв в строках. Это функция логического типа, работает на основе сравнивания в цикле гласных букв и символов в строке.



* 1. Затем нужно узнать содержимое данного нам файла. Для этого используется массив типа string, после объявления которого с помощью стандартного цикла for содержимое файла присваивается в массив и выводится в консоль.



Блок—схема программы

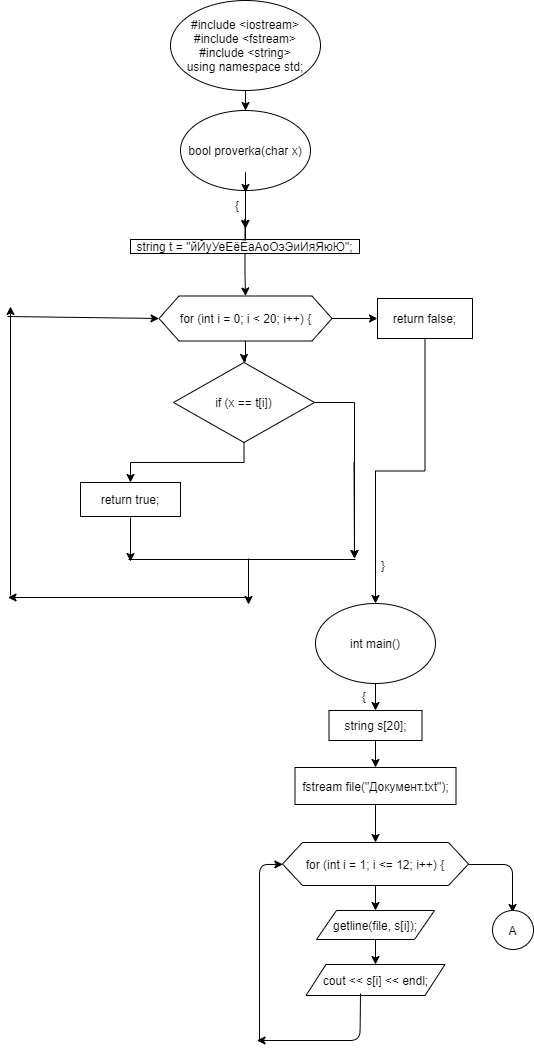


Рисунок 1. Блок-схема программы

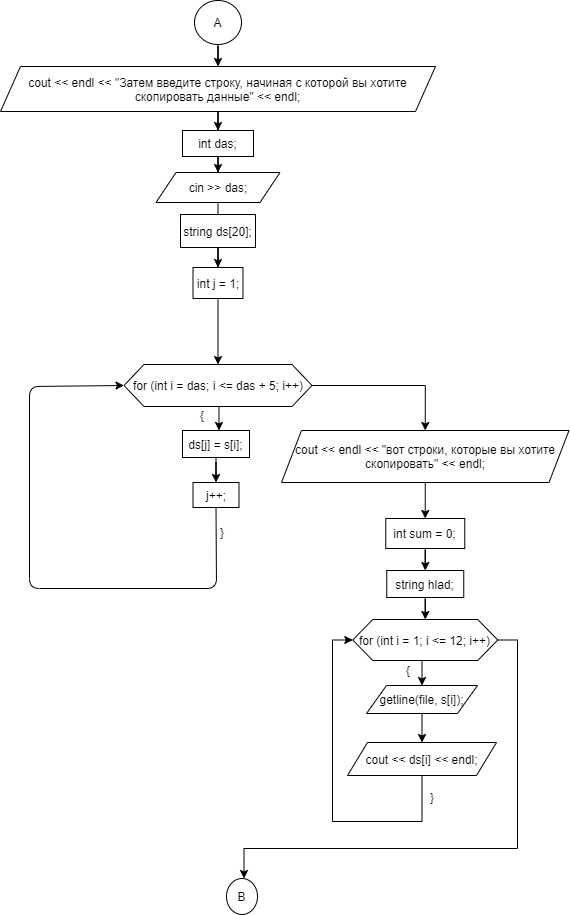


Рисунок . Блок-схема программы (продолжение)

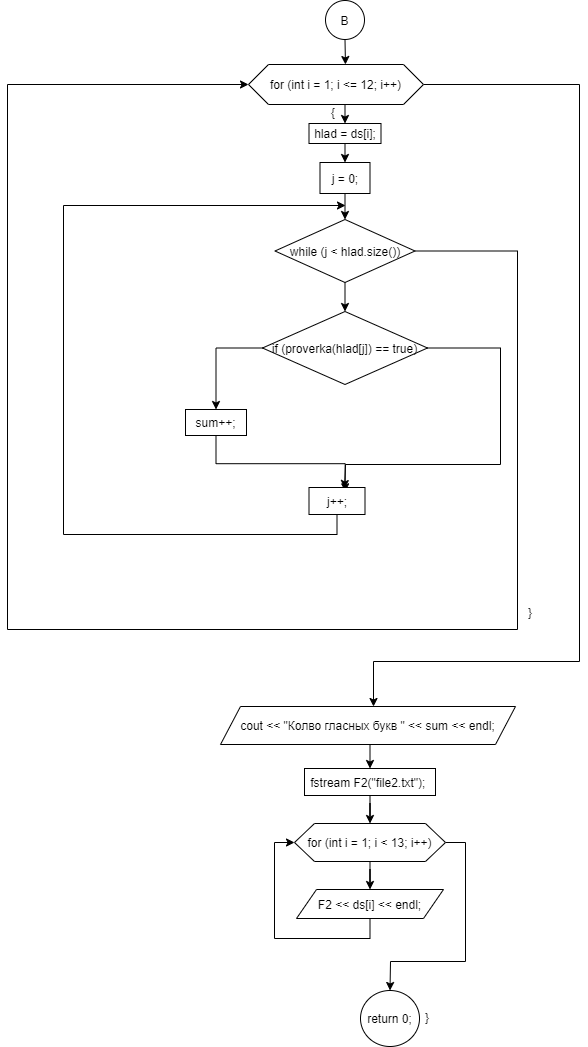


Рисунок . Блок-схема программы (окончание)

Решение

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

bool proverka(char x)

{

string t = "йЙуУеЕёЁаАоОэЭиИяЯюЮ";

for (int i = 0; i < 20; i++)

{

if (x == t[i])return true;

}

return false;

}

int main() {

system("chcp 1251>nul");

setlocale(LC\_ALL, "RUS");

fstream file("Документ.txt");

if (!file)

{

cout << "Файл не открыт\n\n";

return -1;

}

else

{

cout << "Все ОК! Файл открыт!\n\n";

}

string s[20];

for (int i = 1; i <= 12; i++) {

getline(file, s[i]);

cout << s[i] << endl;

}

cout << endl << "Затем введите строку, начиная с которой вы хотите скопировать данные" << endl;

int das;

cin >> das;

while (das <= 0) {

cout << "Такой строки нет, ты ввел строку отрицательную либо нулевую строку";

cin >> das;

}

while (das + 5 >= 13) {

cout << "Вы ввели такое значение, что вылетаете за границы массива...";

cin >> das;

}

string ds[20];

int j = 1;

for (int i = das; i <= das + 5; i++) {

ds[j] = s[i];

j++;

}

cout << endl << "вот строки, которые вы хотите скопировать" << endl;

int sum = 0;

string hlad;

for (int i = 1; i <= 12; i++) {

getline(file, s[i]);

cout << ds[i] << endl;

}

for (int i = 1; i <= 12; i++) {

hlad = ds[i];

j = 0;

while (j < hlad.size()) {

if (proverka(hlad[j]) == true) {

sum++;

}

j++;

}

}

cout << endl << "Колво гласных букв" << sum << endl;

fstream F2("file2.txt");

for (int i = 1; i < 13; i++) {

F2 << ds[i] << endl;

}

if (!F2)

{

cout << "Файл не открыт\n\n";

return 0;

}

else

{

cout << "Все ОК! Файл открыт!\n\n";

}

}

Скриншоты

