Выполнила

студентка группы КТбо1-2 Н. А. Свидич

Принял

ассистент ИКТИБ Д. С. Кочубей

Таганрог 2023

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

(ФГАОУ ВО «ЮФУ»)

Институт компьютерных технологий и информационной безопасности

Кафедра системного анализа и телекоммуникаций

**ЛАБОРАТОРНАЯ РАБОТА № 4**

по дисциплине «Основы алгоритмизации и программирования»

Вариант 2

на тему:

«Файлы данных»

**Содержание**

Содержание 2

Техническое задание 3

Цель работы 3

Задача 3

Ход работы 4

Алгоритм 4

Блок-схема 5

Написание программы 6

Примеры работы программы 7

Вывод 8

Листинг (Приложение а) 9

**Техническое задание**

**Цель работы**

Цель данной лабораторной работы ознакомиться с понятием файла данных, научиться правильно использовать это понятие в программах.

**Задача**

Вариант 2.

Написать программу, которая во вводимом в исходном файле тексте выберет слова, начинающиеся с гласной буквы, и запишет их в другой текстовый файл. Входной текст не ограничивается одной строкой.

Ввод: Файл текстового формата с набором слов

Вывод: Файл текстового формата с полученным через фильтрацию набором слов

**Ход работы**

**Алгоритм**

Для решения задачи алгоритм действия имеет следующие шаги:

1. Получение строки из исходного файла. Посимвольное чтение строки до символа завершения ‘\0’ или до заполнения буфера хранения строки в 256 символов.
2. Проверка символа на наличие гласной буквы и начала слова.
3. В случае выполнения условия: посимвольный вывод слова в конечный файл.
4. Переход к чтению следующего слова из исходного файла.

**Блок-схема**

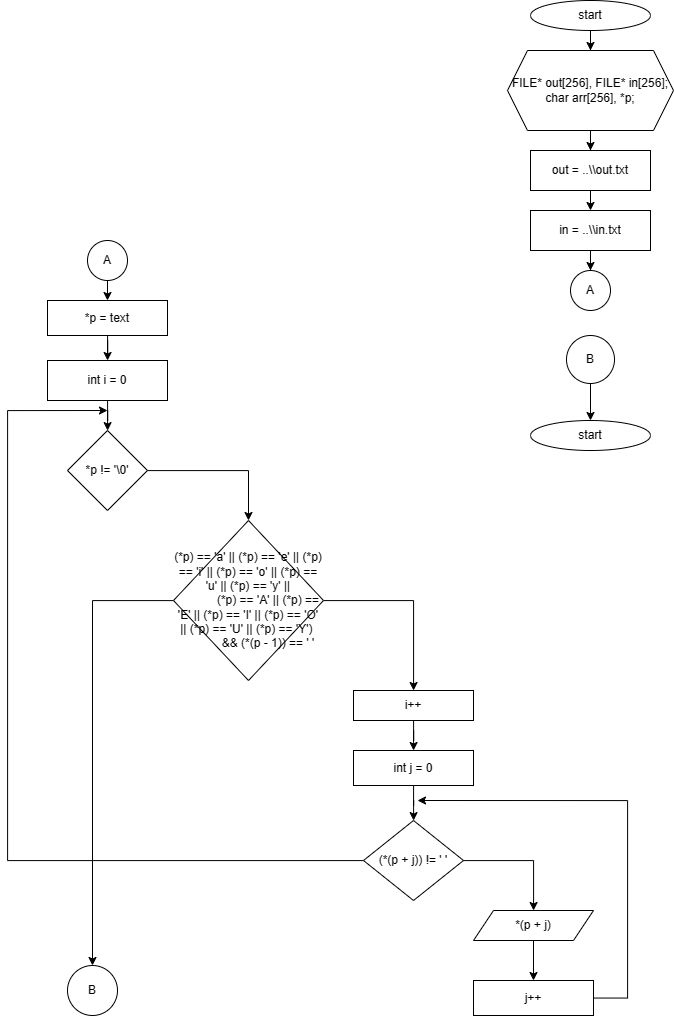


Рисунок 1 – Блок-схема поэтапного решения задачи

**Написание программы**

1. Определение и инициализация переменных.

Так как необходимо считывание и запись в файлы данных, используются переменные типа «FILE\*», в которых хранятся адреса-ссылки на используемые файлы:

Out – хранит ссылку на исходный файл.

In – хранит ссылку на файл для записи полученных данных.

При написании программы была написана функция, имеющая в себе передаваемые параметры типа «FILE\*» - «out», «in»; переменные типа «char» - массив «arr», хранящий 256 символов считанной с файла строки и указатель на считанную строку «\*p».

1. Основные процессы.

Необходимые файлы программа получает через функцию «fopen()», которая имеет параметры с адресом файлов и методом их обработки.

Главная задача вынесена в отдельную функцию «transfers», которая получает, обрабатывает и записывает строки файлов.

В первую очередь, исходный файл проверяется на корректное открытие и в случае ошибки программа выдает предупреждение. Затем, программа начинает чтение файла, пока не дойдет до его конца.

Далее, поочередно считывается каждая строка до конечного символа или до заполнения буфера хранения (массив «arr» длиной 256 символов).

Строка посимвольно перебирается циклом «for», который увеличивает значение указателя \*p для продвижения по строке и завершается, когда считываемый символ равен конечному символу строки.

Во время цикла происходит проверка, является ли символ гласной буквой и одновременно первой буквой слова.

При выполнении условия, с помощью цикла «for», применяющим индекс «j» с пошаговым увеличением и заканчивающимся, когда значения индекса и указателя будут в сумме указывать на символ пробела, обозначая конец слова, происходит посимвольная запись данных в конечный файл.

Иначе, цикл начинает перебор следующих символов.

После выполнения работы функции, открытые ранее файлы закрываются командой «fclose()», для освобождения памяти.

На рисунке 2 показан результат работы полученного программного кода.

**Примеры работы программы**

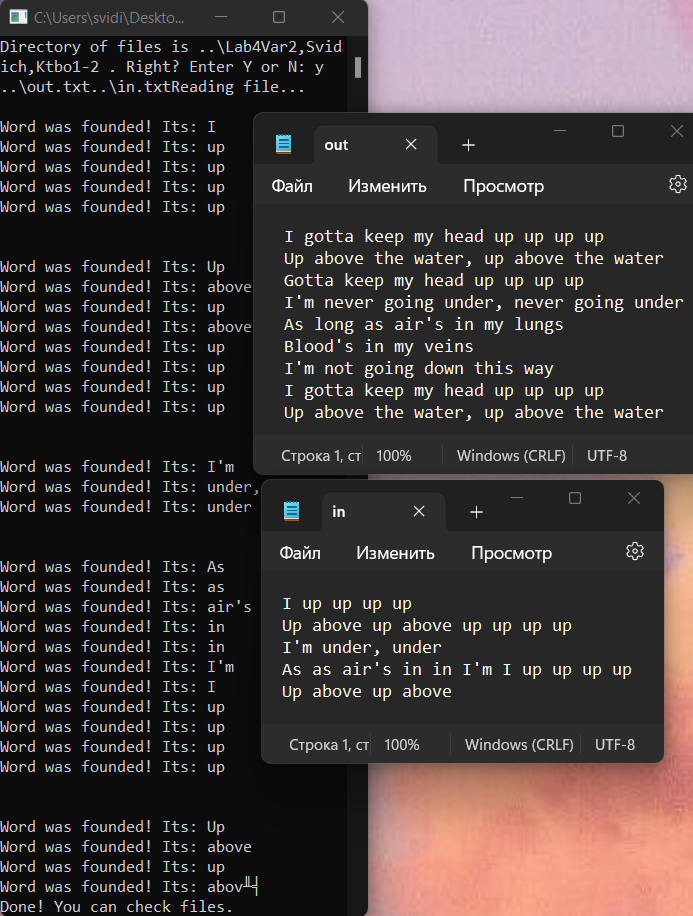


Рисунок 2 –Выполнение программного кода

**Вывод**

Ознакомились с понятием файла данных, научились правильно использовать это понятие в программах.

# **Листинг (Приложение а)**

#include <iostream>

#include <stdio.h>

// Функция, выполняющая считывание строк с файла, проверку наличия в них слов с гласной буквы и их вывод в другой файл

void transfers(FILE\* out, FILE\* in) {

char arr[256]; // Массив, хранящий считываемую строку, до 256 символов

char\* p; // Указатель на считываемую строку

// Проверка на корректное открытие файла

if (out == NULL) perror("Ошибка открытия файла");

else

{

// Проверка на конец файла

while (!feof(out))

{

// Считывание строки и проверка на конец или наличие считанных 255 символов

while ( (p = fgets(arr, 256, out)) != NULL) {

// Начало перебора строки до последнего символа

for (p; (\*p) != '\0'; (p++)) {

// Проверка слова на начальную гласную букву

if (

((\*p) == 'a' || (\*p) == 'e' || (\*p) == 'i' || (\*p) == 'o' || (\*p) == 'u' || (\*p) == 'y' ||

(\*p) == 'A' || (\*p) == 'E' || (\*p) == 'I' || (\*p) == 'O' || (\*p) == 'U' || (\*p) == 'Y')

&& (\*(p - 1)) == ' '

) {

// Цикл для вывода слова, начинающегося на гласную

for (int j = 0; (\*(p + j)) != ' '; j++) {

fputc(\*(p + j), in);

// Разделение слов при выводе

if (\*(p + j + 1) == ' ') fputc(' ', in);

// Прекращение цикла вывода слова, если был выведен последний символ последнего слова

if (\*(p + j + 1) == '\0') break;

}

}

}}}}

}

int main()

{

// Объявление файлов и пути их хранения

FILE\* out = fopen("..\\out.txt", "r");

FILE\* in = fopen("..\\in.txt", "w");

// Использование функции для вывода в файл in слов из файла out

transfers(out, in);

// Закрытие файлов для освобождение памяти

fclose(out);

fclose(in);

return 0;

}