Министерство науки и высшего образования РФ

Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа киберфизических систем и управления

УДК 004.421

УТВЕРЖДАЮ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г.

Отчет

по дисциплине «Теория и технология программирования»

Лабораторная работа №3 «Символьные строки», вариант 14

Выполнил:

Студент

гр.3530902/90001 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Непушкин С.А. подпись, дата

Доцент ВШКФСиУ,

Кандидат технических наук

С. В. Хлопин \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись, дата

Санкт-Петербург

2019

**1. Цель работы.**

Цель лабораторной работы состоит в формировании умений:

- использовать различные способы описания и формирования символьных строк;

- знать и уметь пользоваться методами чтения и записи строк в текстовых файлах;

- знать и уметь пользоваться методами чтения и анализа потоковых данных, вводимых с клавиатуры.

**2. Задание:**

Написать программу, которая предлагает пользователю ввести текст. Программа должна для каждой строки вычислить среднее арифметическое значений длин слов в строке.

**3. Текст программы**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include<stdio.h>

#include<stdlib.h>

#include<locale.h>

int isDigit(char c)

{

if (c >= '0' && c <= '9')

{

return 1;

}

return 0;

}

char charToLower(char c)

{

if (c >= 'A' && c <= 'Z')

{

c = c - 'A' + 'a';

}

else

{

if (c >= 'А' && c <= 'Я')

{

c = c - 'А' + 'а';

}

}

if (c == '\\') return '/';

return c;

}

int lowerstrcmp(char\* str1, char\* str2)

{

int i = 0;

while (str1[i] != '\0' && str2[i] != '\0' && charToLower(str1[i]) == charToLower(str2[i]))

{

i++;

}

return (charToLower(str1[i]) - charToLower(str2[i]));

}

int lowerstartswith(char\* str1, char\* str2)

{

int i = 0;

while (str1[i] != '\0' && str2[i] != '\0' && charToLower(str1[i]) == charToLower(str2[i]))

{

i++;

}

if (str2[i] == '\0')

{

return 1;

}

return 0;

}

int isLegalFileName(char\* str)

{

if (lowerstrcmp(str, "con") == 0 || lowerstartswith(str, "con.") ||

lowerstrcmp(str, "nul") == 0 || lowerstartswith(str, "nul.") ||

lowerstrcmp(str, "prn") == 0 || lowerstartswith(str, "prn.") ||

lowerstrcmp(str, "aux") == 0 || lowerstartswith(str, "aux.") ||

lowerstrcmp(str, "clock$") == 0 || lowerstartswith(str, "clock$."))

{

return 0;

}

if (lowerstartswith(str, "com") || lowerstartswith(str, "lpt"))

{

int number = 0;

int index = 3;

while (str[index] != '\0')

{

if (!isDigit(str[index]))

{

return 1;

}

number = number \* 10 + str[index] - '0';

index++;

}

if (number >= 0 && number <= 32)

{

return 0;

}

}

return 1;

}

int fileExists(char\* str)

{

FILE\* tmp = fopen(str, "r");

if (tmp == 0)

{

return 0;

}

fclose(tmp);

return 1;

}

char\* getFileName(char\* str)

{

int index = 0;

int lastSlash = -1;

while (str[index] != '\0')

{//D:/1\asdf/1asf/12aws\1.txt

if (str[index] == '\\' || str[index] == '/')

{

lastSlash = index;

}

index++;

}

return str + lastSlash + 1;

}

int main()

{

setlocale(LC\_ALL, "Russian");

int line = 1;

int currentlettercount = 0;

int currentwordcount = 0;

float avglength = 0;

int isword = 0;

int state = 1;

/\*

state == 0 -> буква

state == 1 -> не буква

\*/

char chr;

FILE \*inFile;

char inputFile[100];

printf("Введите имя входного файла: ");

gets(inputFile);

printf("Введенное имя входного файла: \"%s\"\n", inputFile);

if (!isLegalFileName(getFileName(inputFile)))

{

printf("Некорректное имя входного файла : \"%s\"", getFileName(inputFile));

return 0;

}

if ((inFile = fopen(inputFile, "r")) == 0)

{

printf("Некорректное имя входного файла");

return 0;

}

FILE \*outFile;

char outputFile[100];

printf("Введите имя выходного файла: ");

gets(outputFile);

printf("Введенное имя выходного файла: \"%s\"\n", outputFile);

if (!isLegalFileName(getFileName(outputFile)))

{

printf("Имя файла зарезервировано, пожалуйста, введите другое имя");

return 0;

}

if (lowerstrcmp(outputFile, inputFile) == 0)

{

printf("Имя входного файла совпадает с именем выходного файла");

return 0;

}

if (fileExists(outputFile))

{

if ((outFile = fopen(outputFile, "w")) == 0)

{

printf("Невозможно открыть файл для записи, файл доступен для чтения");

return 0;

}

}

else

{

if ((outFile = fopen(outputFile, "w")) == 0)

{

printf("Некорректное имя выходного файла");

return 0;

}

}

while (chr = fgetc(inFile)) {

fprintf(outFile, "%c", (chr == EOF ? '\n' : chr));

switch (chr)

{

case ' ': case '\t': case '.': case ',':

case '-': case '!': case'?':

if (state == 0) {

currentwordcount++;

}

state = 1;

break;

case '\n': case EOF:

if (state == 0) {

currentwordcount++;

}

avglength = (currentwordcount == 0 ? 0 : (float)currentlettercount / currentwordcount); //запись переменной в float

fprintf(outFile, "%d line -> %4.2f\n", line, avglength);

state = 1;

line++;

currentwordcount = 0;

currentlettercount = 0;

break;

default:

currentlettercount++;

state = 0;

break;

}

if (chr == EOF) {

break;

}

}

fclose(inFile);

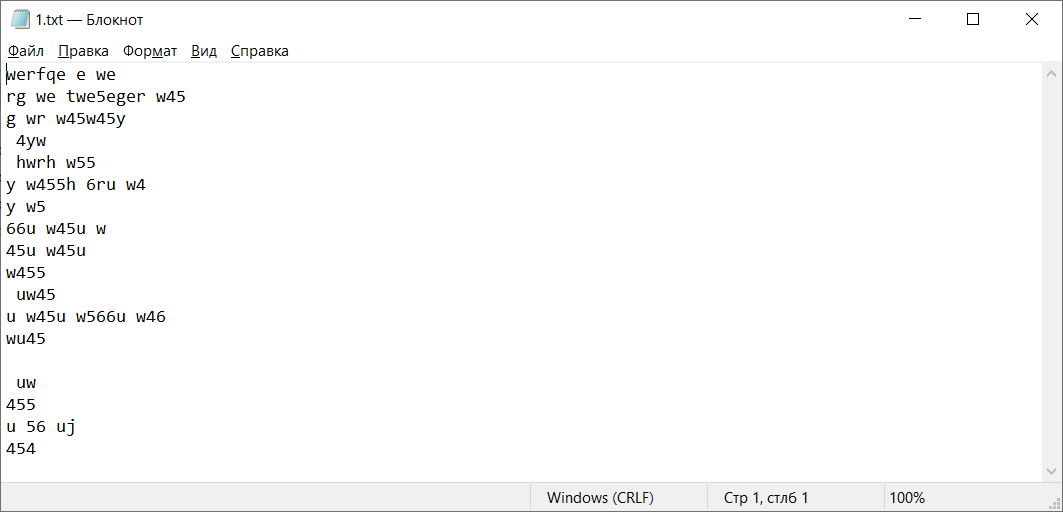
fclose(outFile);

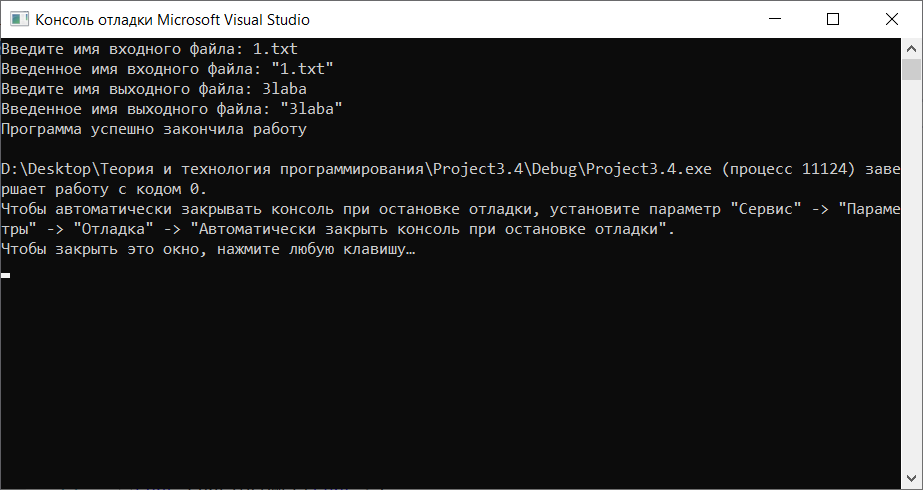
printf("Программа успешно закончила работу\n");

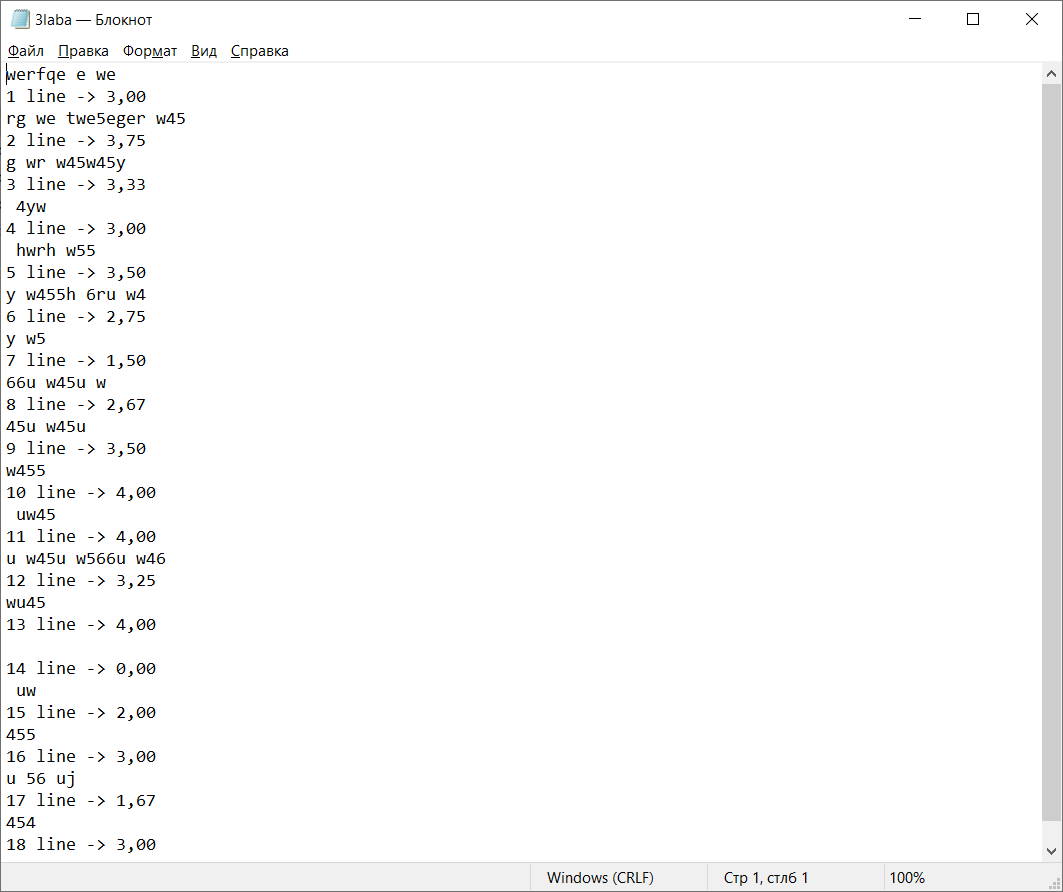
return 0;

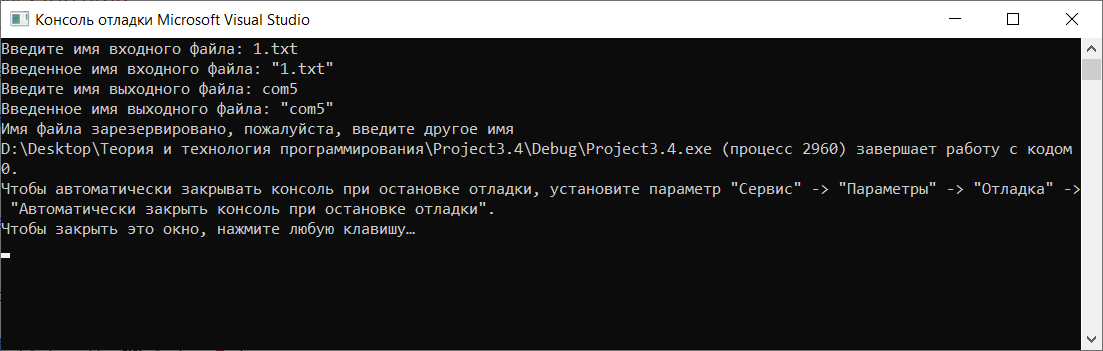
}

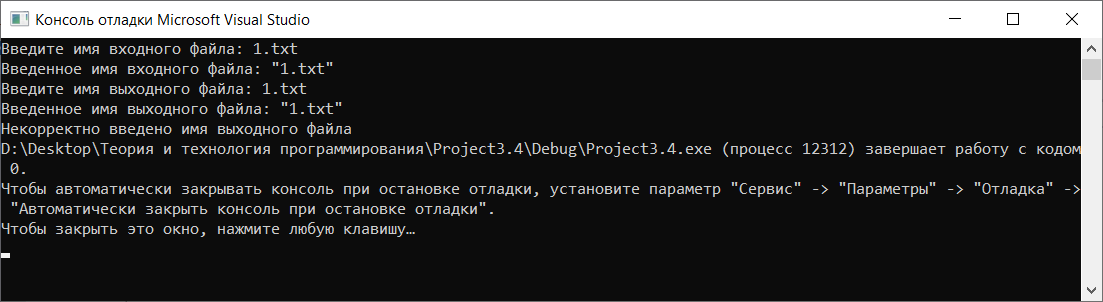
**4. Пример работы программы**











**Вывод**

В ходе лабораторной работы я:

-научился выводить и вводить данные из файла

-прочитал об индикаторе «EOF»

-научился работать с оператором выбора «switch», а также с операторами «case» и «default»

-вводить оператор выбора «if» в одну строчку

-узнал о зарезервированных именах MS DOS

-научился проверять название файла на корректный ввод

-прочитал о функции tolower

-прочитал о функции strcmp