

Моисеенко Андрей Сергеевич Обработка текстовой информации



Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)
Институт новых материалов и технологий
Кафедра Информационных технологий и автоматизации проектирования

Оценка _____

Руководитель курсового
проектирования _____

Члены комиссии _____

Дата защиты _____

ОТЧЕТ

о курсовой работе по дисциплине «Алгоритмизация и программирование»
по теме: «Обработка текстовой информации»

Вариант № 19

Студент: Моисеенко Андрей Сергеевич
(ФИО)

(Подпись)

Группа: НМТ-113929

Екатеринбург
2022 г.



**Уральский
федеральный
университет**

имени первого
Президента России
Б.Н. Ельцина

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)
Институт новых материалов и технологий
Кафедра Информационных технологий и автоматизации проектирования

Задание на курсовую работу

Студент: Моисеенко Андрей Сергеевич

группа: НМТ -113929

направление подготовки: 09.03.02 «Информационные системы и технологии»

1. Тема курсовой работы: «Обработка текстовой информации», вариант № 35.
2. Содержание работы, в том числе состав графических работ и расчетов
должен быть обработан текстовый файл, найдено заданное с клавиатуры
СЛОВО

(какие графические работы и расчёты должны быть выполнены)

3. Дополнительные сведения

4. План выполнения курсовой работы

Наименование элементов проектной работы	Сроки	Примечания	Отметка о выполнении
Алгоритм обработки информации	16.04.2022-23.04.2022		
Ввод информации, вывод ее на экран	24.05.2022-09.05.2022		
Тестирование главной программы	09.05.2022-16.05.2022		
Оформление отчета	16.05.2022-23.05.2022		

Руководитель _____ /Сердюк Ю.В./

Содержание

Введение.....	4
Отчет о решении поставленной задачи.....	4
1. <i>Постановка задачи</i>	4
2. Алгоритм.....	4
3. Текст программы.....	7
4. Результат работы программы.....	8
Заключение.....	8

Введение

Цель курсовой работы является обучение работы с текстовыми файлами, правильной обработкой информации и закрепление знаний основ алгоритмизации и умений решать с помощью персонального компьютера типовых инженерно-технических задач программными методами с использованием языка Си. В курсовой работе необходимо обработать текстовый файл, считать информацию из него и найти заданное слово.

Отчет о решении поставленной задачи

1. Постановка задачи

Вариант №19

С помощью текстового редактора создать файл, содержащий произвольный текст с латинскими символами. Имя файла должно иметь расширение txt. Написать программу, которая:

- выводит текст на экран дисплея;
- по нажатию произвольной клавиши поочередно выделяет каждое из слов текста, у которых первый и последний символы совпадают;
- Выводит текст на экран дисплея еще раз, заменяя в заданном слове строчные буквы прописными.

Алгоритм

1. Подключение библиотек
2. Описание прототипов подпрограмм
3. Описание переменных
4. Считывание текста из файла
 1. Ввод имени файла

```
printf("Введите полное имя файла\n");
scanf("%s",name);
f_in=fopen(name,"rb");
```
 2. Проверка на возможность открытия файла

```
if(f_in==NULL)
{ printf("Невозможно открыть файл %s.",name);
  getch();
  exit(1);};
```

3. Перемещение указателя относительно конца файла
(fseek(f_in,0,SEEK_END))
4. Нахождение длины текста (l_str=ftell(f_in))
5. Проверка на наличие текстовой информации
if(l_str==0)
{ printf("\nВ файле %s отсутствует текст!",name);
 getch();
 exit(1);}
6. Выделение памяти для хранения символьного массива
 S=(char*)malloc(l_str*sizeof(char));
7. Перемещение указателя относительно начала файла
 fseek(f_in,0,SEEK_SET);
8. Запись информации из файла в символьный массив
 for(i=0;i<l_str;i++)
 fscanf(f_in,"%c",S+i);
 *(S+l_str)='\0';
5. Вывод исходного текста на экран
 1. Очистка экрана (clrscr())
 2. Вывод исходного текста на экран
printf("Исходный текст:\n");
for(i=0;i<l_str;i++)
{ if (i%79==0)
 printf("\n");
 putch(*(S+i));}
6. Вывод количества слов в тексте
 1. Вызов подпрограммы для проверки наличия текста
 Sign (S,l_str,&error); где S – символьный массив содержащий текст;
 l_str– длина символьного массива; p – «флаг», который показывает, есть ли предложения
 2. Вызов подпрограммы для поиска слов, у которых первый и последний символ совпадает
 Count_Words(S,l_str,&k); где S – символьный массив содержащий текст;
 l_str– длина символьного массива; k – количество слов, у которых первых и последний символ совпадает

7. Вывод текста с выделенным словом

1. Организуется цикл `while (from+1<l_str)` (позиция указателя плюс 1 меньше длины символьного массива)

a) Очистка экрана (`clrscr()`)

b) Вызов подпрограммы для нахождения начала и конца слов, у которых первый и последний символ совпадает

c) Организуется условный цикл (`if ((*S+begin)==*(S+end)) &&(begin!=end))`)

1. Вызов подпрограммы для печати в нормально режиме до начала нахождения слов с одинаковым первым и последним символом

2. Вызов подпрограмм для выделения цветом найденного цвета

3. Вызов подпрограммы для печати до конца текста

d) Иначе `break`

2. Очистка экрана

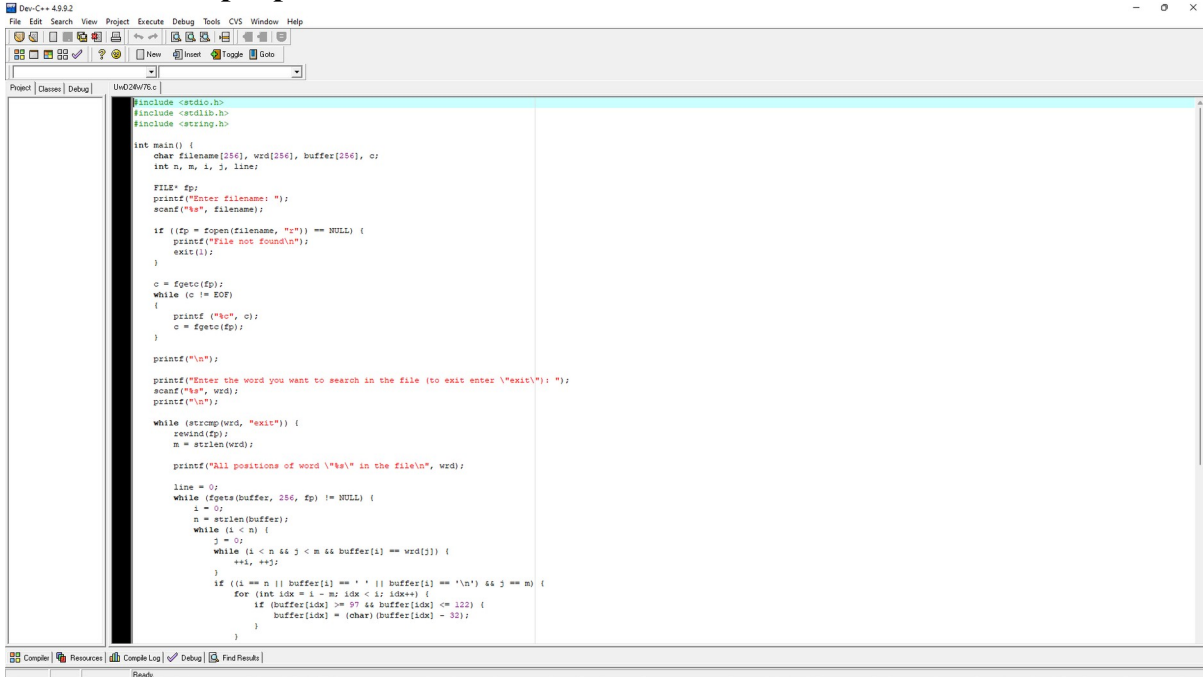
3. Вывод исходного текста

```
for(i=0;i<l_str;i++)  
{ if (i%79==0)  
    printf("\n");  
    putchar(*(S+i));}
```

8. Освобождение памяти

9. Тексты подпрограмм

2. Текст программы



```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main() {
    char filename[256], wrd[256], buffer[256], c;
    int n, m, i, j, line;

    FILE* fp;
    printf("Enter filename: ");
    scanf("%s", filename);

    if ((fp = fopen(filename, "r")) == NULL) {
        printf("File not found\n");
        exit(1);
    }

    c = fgetc(fp);
    while (c != EOF) {
        printf("%c", c);
        c = fgetc(fp);
    }

    printf("\n");
    printf("Enter the word you want to search in the file (to exit enter \"exit\"): ");
    scanf("%s", wrd);
    printf("\n");

    while (strcmp(wrd, "exit") != 0) {
        rewind(fp);
        m = strlen(wrd);

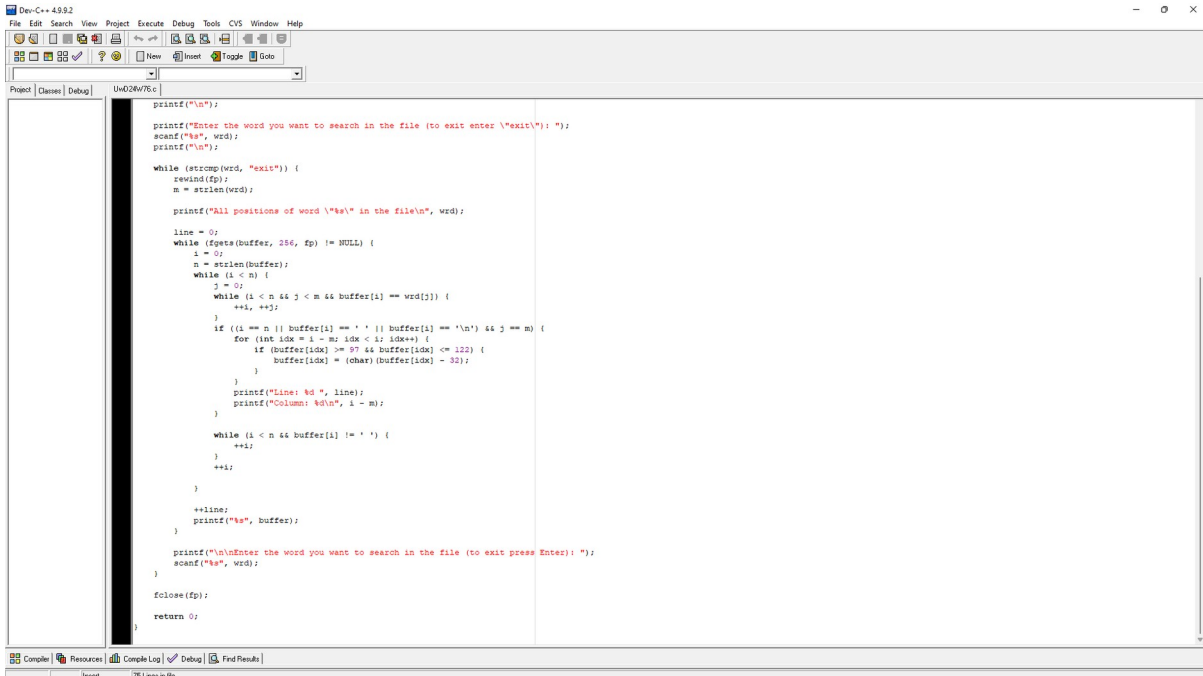
        printf("All positions of word \"%s\" in the file\n", wrd);

        line = 0;
        while (fgets(buffer, 256, fp) != NULL) {
            i = 0;
            n = strlen(buffer);
            while (i < n) {
                j = 0;
                while (i < n && j < m && buffer[i] == wrd[j]) {
                    ++i, ++j;
                }
                if ((i == n || buffer[i] == ' ' || buffer[i] == '\n') && j == m) {
                    for (int idx = i - m; idx < i; idx++) {
                        if (buffer[idx] >= '0' && buffer[idx] <= '9')
                            buffer[idx] = (char)(buffer[idx] - 32);
                    }
                }
                printf("Line: %d ", line);
                printf("Column: %d\n", i - m);
            }
            ++i;
        }
        ++line;
        printf("%s", buffer);
    }

    printf("\nEnter the word you want to search in the file (to exit press Enter): ");
    scanf("%s", wrd);
}

fclose(fp);
return 0;
```

Рисунок 1



```
printf("\n");
printf("Enter the word you want to search in the file (to exit enter \"exit\"): ");
scanf("%s", wrd);
printf("\n");

while (strcmp(wrd, "exit") != 0) {
    rewind(fp);
    m = strlen(wrd);

    printf("All positions of word \"%s\" in the file\n", wrd);

    line = 0;
    while (fgets(buffer, 256, fp) != NULL) {
        i = 0;
        n = strlen(buffer);
        while (i < n) {
            j = 0;
            while (i < n && j < m && buffer[i] == wrd[j]) {
                ++i, ++j;
            }
            if ((i == n || buffer[i] == ' ' || buffer[i] == '\n') && j == m) {
                for (int idx = i - m; idx < i; idx++) {
                    if (buffer[idx] >= '0' && buffer[idx] <= '9')
                        buffer[idx] = (char)(buffer[idx] - 32);
                }
            }
            printf("Line: %d ", line);
            printf("Column: %d\n", i - m);
        }
        ++i;
    }
    ++line;
    printf("%s", buffer);
}

printf("\nEnter the word you want to search in the file (to exit press Enter): ");
scanf("%s", wrd);
}

fclose(fp);
return 0;
```

Рисунок 2

3. Результат работы программы

```
Enter filename: test.txt
Enter the word you want to search in the file (to exit enter "exit"): John
All positions of word "John" in the file
Line: 0 Column: 17
Hello My name is JOHN Smith
Line: 1 Column: 0
Line: 1 Column: 20
JOHN is a Good man. JOHN is very Tall.
Line: 2 Column: 20
Please stop talking JOHN

Enter the word you want to search in the file (to exit press Enter): is
All positions of word "is" in the file
Line: 0 Column: 14
Hello My name IS John Smith
Line: 1 Column: 5
Line: 1 Column: 25
John IS a Good man. John IS very Tall.
Please stop talking John

Enter the word you want to search in the file (to exit press Enter): exit
```

Рисунок 3

На фото показано название файла, слово которое мы выделяем “John” и хотим заменить буквы с строчных на прописные.

Заключение

В курсовой работе был обработан текстовый файл, в выбранных нами словах буквы были заменены с строчных на прописные. Программа корректно обрабатывает информацию, выводит исходный текст, выделяет цветом слова и подсчитывает их количество. В ходе выполнения работы были отработаны навыки работы с текстовыми документами и написанием подпрограмм.