**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра Вычислительной техники**

**ОТЧЕТ**

**по лабораторной работе № 5**

**по дисциплине «Программирование»**

**Тема: «Обработка строк»**

| Студент гр. 3311 |  | Сапронов К. Д. |
| --- | --- | --- |
| Преподаватель |  | Хахаев И. А. |

Санкт-Петербург

2023

**Цель работы.**

Целью работы является изучение строк и получение практических навыков в их обработке.

**Задание (вариант 6)**

Ввести строку символов-разделителей между словами, а затем строку текста, состоящую из слов и символов-разделителей, находящихся в произвольном количестве до и после слов. Подсчитать и вывести количество слов, превышающих заданное количество символов.

**Постановка задачи и описание решения**

Для выполнения данной задачи необходимо написать программу, которая после ввода всех данных будет вносить слова из текста в файл, после чего внутри файла будет проверять длину каждого слова и подсчитывать количество слов, превышающих заданную длину.

1. Инициализируются следующие переменные: seps и text - строки разделителей и текста, имеющие большой размер, lim - задаваемое количество символов, \*filename - указатель на имя файла
2. Создаются следующие функции: sepcheck(char ch, char \*seps) - проверяет, является ли данный символ разделителем; addwordstofile(char \*str, const char \*filename, char \*seps) - добавляет строку текста в файл так, что каждая строка в файле - это отдельное слово; countwords(const char \*filename, char \*seps, int lim) - подсчитывает в файле слова, превышающие заданную длину, после чего закрывает файл
3. С клавиатуры вводятся строки разделителей и текста, а также задается количество символов в слове
4. С помощью функции addwordstofile текст заносится в файл. Эта функция работает следующим образом: она создает строку, которая заполняется символами из строки текста. Функция посимвольно проверяет текст функцией sepcheck, и если символ является разделителем, то созданная строка записывается в файл с переносом строки. После этого счетчик для созданной строки обнуляется, за счет чего она начинается перезаписываться с начала при записи нового слова
5. С помощью функции countwords подсчитывается и выводится количество слов, удовлетворяющих условию

**Описание переменных**

main()

| № | Имя переменной | Тип | Назначение |
| --- | --- | --- | --- |
| 1 | seps | char | Хранение строки разделителей |
| 2 | text | char | Хранение строки текста |
| 3 | lim | int | Задаваемая длина слова |
| 4 | filename | char | Имя файла |

sepcheck()

| № | Имя переменной | Тип | Назначение |
| --- | --- | --- | --- |
| 1 | ch | char | Проверяемый символ |
| 2 | seps | char | Строка разделителей |

addwordstofile()

| № | Имя переменной | Тип | Назначение |
| --- | --- | --- | --- |
| 1 | str | char | Строка текста |
| 2 | filename | char | Имя файла |
| 3 | seps | char | Строка разделителей |
| 4 | line | char | Строка для хранения текущего слова |
| 5 | index | int | Номер символа текущего слова |
| 6 | i | int | Переменная в цикле |
| 7 | file | char | Текстовый файл, в который записываются слова |

countwords()

| № | Имя переменной | Тип | Назначение |
| --- | --- | --- | --- |
| 1 | filename | char | Имя файла |
| 2 | seps | char | Строка разделителей |
| 3 | lim | int | Заданная длина слова |
| 4 | count | int | Количество слов, удовлетворяющих условию |
| 5 | line | char | Строка, содержащая текущее слово (строку файла) |
| 6 | file | char | Текстовый файл со словами |

**Контрольные примеры**

**Пример 1:**

Ввод: разделители: пробел; текст: a ab abc abcd; длина: 2;

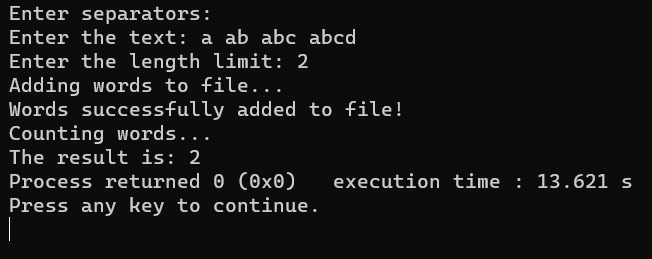
Вывод: 2

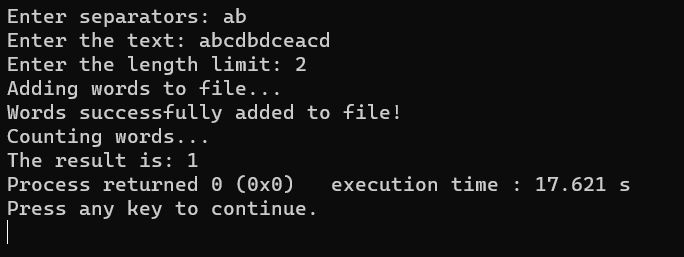
**Пример 2:**

Ввод: разделители: ab; текст: abcdbdceacd; длина: 2;

Вывод: 1

**Примеры выполнения программы**

****

****

**Выводы.**

В результате выполнения работы изучены строки и получены практические навыки в их обработке.