**Отчет по лабораторной работе № 11** по курсу “Фундаментальная информатика”

Студент группы М80-101Б-22, Бычков Артур Сергеевич, № по списку 2

Контакты email: bychkovarthur@gmail.com

Работа выполнена: «20» ноября 2022г.

Преподаватель: каф. 806 Крылов Сергей Сергеевич

Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Отчет сдан « » \_\_\_\_\_\_\_\_\_202 \_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Тема:** Обработка последовательности литер входного текстового файла. Простейшие приемы лексического анализа. Диаграммы состояний и переходов
2. **Цель работы:** Составить программу на языке Си, выполняющую анализ и обработку вводимого текста

1. **Задание** (вариант № 2)**:** Подсчитать число слов в многострочных комментариях (/\* и \*/) в программе на Си.

1. **Оборудование**:

*Оборудование ПЭВМ студента, если использовалось:*

Процессор **AMD Ryzen 5 5500U 2.10 GHz, 6 ядер** с ОП **8192 Мб**, ТТН **512000 Мб**. Мониторы **Lenovo.**

1. **Программное обеспечение:**

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства **Linux**, наименование **Ubuntu** версия **20.04.5,** интерпретатор команд bash версия 5.0.17(1).

Система программирования CLion версия 2021.1.3

Редактор текстов nano версия 6.2

Утилиты операционной системы WinRar, Microsoft Word.

Прикладные системы и программы Ubuntu wsl, Clion, Google Chrome

Местонахождение и имена файлов программ и данных на домашнем компьютере /home/artur

1. **Идея, метод, алгоритм** решения задачи(в формах:словесной,псевдокода,графической[блок-схема,диаграмма,рисунок,таблица] или формальные спецификации с пред- и постусловиями)
   1. Будем обрабатывать только слова, находящиеся в комментарии. Для этого создадим 2 состояния, c и in\_word. Если первое равно 1, то мы находимся в многострочном комментарии, иначе вне него. Если in\_word равен 1, то мы внутри слова (слово в этом задании любая непустая последовательность символов, не являющихся разделителями).
   2. В состоянии с = 0, мы ищем момент, когда текущий символ это “\*”, а предыдущий – это “/”. В этом случае переключаем с на 1 и со следующего шага мы находимся внутри комментария и считаем количество слов.
   3. В состоянии in\_word = 0 мы ищем то, когда текущий символ НЕ является символом – разделителем (пробел, табуляция, перенос строки) при помощи функции isspace(symbol). Если текущий символ – не разделитель, то мы переключаем состояние in\_word в положение 1 и теперь мы находимся внутри слова.
   4. В состоянии in\_word = 1 мы ищем то, когда текущий символ является разделителем. Если текущий символ – разделитель, то слово закончилось, мы увеличиваем счетчик на единицу, а само состояние сбрасываем до нуля.
   5. В состоянии с = 1, мы ищем ждем момента, когда текущий символ – это “/”, а предыдущий – “\*”. Если это произошло, то комментарий закончился и мы вновь сбрасываем с до нуля, то есть с = 0.
   6. Однако может произойти такое, что комментарий закончился, однако последнее слово от конца комментария не отделялось пробелом. Например, строка “/\* string string string\*/”. В этом случае программа такой алгоритм найдет только первые 2 слова, а последнее пропустит. Поэтому, когда мы проделываем действия, указанные в пункте 5. Мы проверяем, является ли предпоследний символ не пробельным и если он не является таковым (то есть, это не пробел, табуляция, символ перевода строки), то мы добавляем в счетчик единицу.

**7. Сценарий выполнения работы** [план работы,первоначальный текст программы в черновике(можно на отдельном листе)итесты либо соображения по тестированию].

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| /\*\*/ Это пустой комментарий, а это /\* не  пустой \*/ | The total number of words located in a multi-line comment in the file lab11\_data.txt is equal to 2. |
| просто текст, ничего особенного /\*проверка      пограничных      случаев\*/, когда слово не отделяется пробелом от концов комментария | The total number of words located in a multi-line comment in the file lab11\_data.txt is equal to 3. |
| /\* \*//\* \*//\*  \*//\*  \*/ четыре пустых комментария | The total number of words located in a multi-line comment in the file lab11\_data.txt is equal to 0. |
| Текст -/\*  зафиксированная на каком-либо материальном носителе человеческая мысль; в\*/  общем плане связная и полная последовательность /\*символов.\*/ | The total number of words located in a multi-line comment in the file lab11\_data.txt is equal to 9 |

**8. Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами,подписанныйпреподавателем).

artur@artur-VirtualBox:~$ cat lab11.c

#include <stdio.h>

#include <stdlib.h>

#include <ctype.h>

/\* Лабораторная работа №11. Вариант №2.

Задание: Подсчитать число слов в многострочных комментариях в программе на Си.

Студент Бычков Артур, группа М8О-101Б-22

Сдано 22.11.22 года\*/

int main() {

char file\_name[1000];

int c = 0, in\_word = 0; // с - означает, в комментарии мы находимся или нет

char s, l, pl, ppl, pppl; // Здесь s (symbol) - текущий символ, l (last) - предыдущий символ, pl (pre\_last) - предпредыдущий символ и последний 2 аналогично.

long long count = 0;

printf("Enter a file name up to 1000 characters: ");

scanf("%s", file\_name);

FILE \*file = fopen(file\_name, "r"); // Открытие файла на чтение данных

while (( s = getc(file)) != EOF) {

if (pppl == '/' && ppl == '\*' && pl == '/' && l == '\*' && s == '/') { // Пограничный случай /\*/\*/

c = 0;

in\_word = 0;

pppl = ppl;

ppl = pl;

pl = l;

l = s;

continue;

}

if (ppl == '/' && pl == '\*' && l == '\*' && s == '/') { // Пограничный случай /\*\*/

c = 0;

in\_word = 0;

continue;

}

if (s == '/' && l == '\*' && c == 1) { // Выходим из комментария

if (!(isspace(pl))) {

count ++;

}

c = 0;

in\_word = 0;

pppl = ppl;

ppl = pl;

pl = l;

l = s;

continue;

}

if (c) { // Мы внутри комментария

if (in\_word == 0 && !(isspace(s))) { // Заходим в слово

in\_word = 1;

}

if (in\_word == 1 && isspace(s)) { // Слово закончислось

in\_word = 0;

count++;

}

pppl = ppl;

ppl = pl;

pl = l;

l = s;

continue;

}

if (l == '/' && s == '\*') { // Заходим в комментарий

c = 1;

pppl = ppl;

ppl = pl;

pl = l;

l = s;

continue;

}

pppl = ppl;

ppl = pl;

pl = l;

l = s;

}

FILE \*ans = fopen("answer.txt", "w"); // Открываем файл на записть ответа

fprintf(ans, "The total number of words located in a multi-line comment in the file %s is equal to %lld\n", file\_name, count); // Ввод ответа в файл

printf("The total number of words located in a multi-line comment in the file %s is equal to %lld\n", file\_name, count); // Вывод ответа в консоль

fclose(file); // Closing files

fclose(ans);

return 0;

}

artur@artur-VirtualBox:~$ gcc lab11.c

artur@artur-VirtualBox:~$ ./a.out

Enter a file name up to 1000 characters: lab11\_data.txt

The total number of words located in a multi-line comment in the file lab11\_data.txt is equal to 4

artur@artur-VirtualBox:~$ cat lab11\_data.txt

/\* Лабораторная работа \*/

№11. Вариант задания/\*№2\*/ выполнил студент /\*\*/Бычков /\*Артур\*/ Сергеевич

группа/М8О-101Б-22.

**9. Дневник отладки** должен содержать дату и время сеансов отладки и основные события(ошибки в сценарии и программе,нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
|  |  |  |  |  |  |  |

1. **Замечания автора** по существу работы

Отсутствуют.

1. **Выводы**

Я научился составлять программы, написанные на Си, для анализа и обработки вводимого текста.

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_