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

Студент группы М80-109Б-22  Фомин Иван Дмитриевич, № 22

Контакты email: [grenka388@gmail.com](mailto:grenka388@gmail.com),

Telegram: @Haliaven

Работа выполнена: «24» ноября 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан «28» \_\_\_11\_\_\_\_2022 г., итоговая оценка \_\_\_\_\_

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

1. **Тема:** Обработка последовательности литер входного текстового файла. Простейшие приёмы лексического анализа. Диаграммы состояний и переходов.

1. **Цель работы:** Составить программу на Си, выполняющую анализ и обработку вводимого текста в соответствии с заданием.

1. **Задание:** Вариант 6. Подсчитать количество слов – правильных идентификаторов языков Pascal и C.

1. **Оборудование** (студента):

Процессор AMD Ryzen 7 4800HS with Radeon Graphics  с ОП 8 Гб, SSD 512 Гб. Монитор *1920x1080*

1. **Программное обеспечение (**студента**):**

Операционная система семейства: *linux*, наименование: *ubuntu*,  версия 22.04*jammy*

интерпретатор команд: *bash* версия 5.1.16(1)-release

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

**6. Идея, метод, алгоритм**

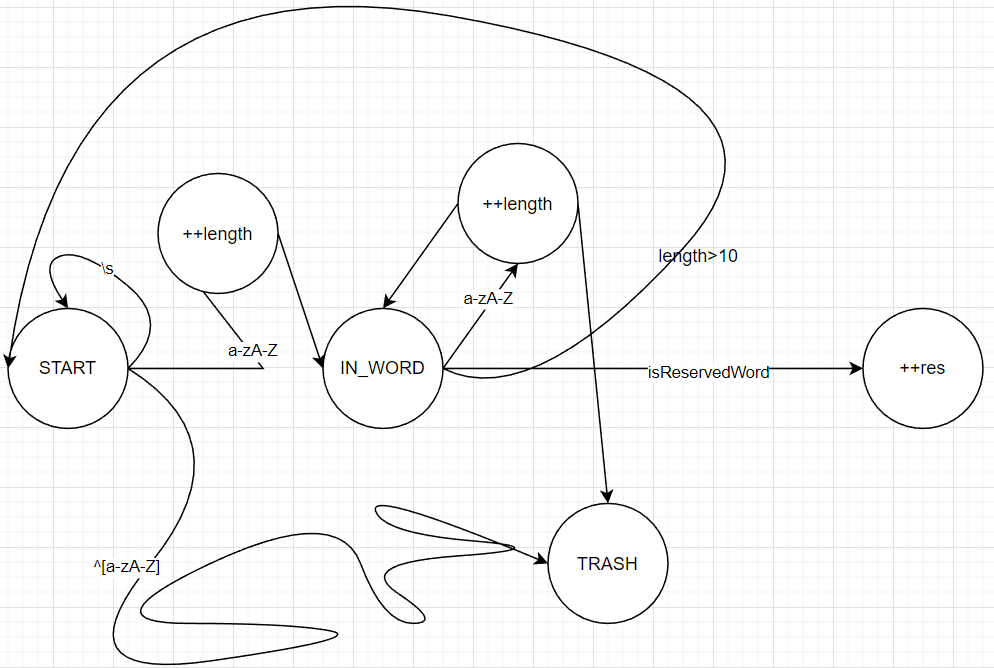
Сначала считываем слово. Словом я называю последовательность НЕ из специальных символов, ограниченная с двух сторон любым символом, который, если вызвать isspace с этим символом в качестве аргумента вернёт положительное число.

Считанное слово проверяем посимвольно через функцию strcmp с зарезервированными словами из Си и Паскаля.

Если слово есть +1, если нет +0.

**7. Сценарий выполнения работы**

|  |  |  |
| --- | --- | --- |
| **Входные данные** | **Выходные данные** | **Описание тестируемого случая** |
| one two then three | 1 | Из всех слов then входит в искомые |
| prog  ram | 1 | Проверка на игнор специальных символов |
| xor 872634 ugefuyg wiuehf case repeat g | 3 | Проверка, если вводятся подходящие слова разной длины |
|  | 0 | Пустой ввод |



**8. Распечатка протокола**

#include <stdio.h>  
#include <assert.h>  
#include <stdlib.h>  
#include <string.h>  
#include <ctype.h>  
  
const char WORD\_LIST[21][10] = {  
 "xor",  
 "case", "else", "then", "lock", "goto", "type",  
 "break", "using",  
 "repeat", "sizeof", "typeof", "static", "inline", "struct", "return", "signed", "extern",  
 "program", "default",  
 "register"  
};  
  
int isReservedWord(int size, char word[size]) {  
 for (int i = 0; i < 21; ++i) {  
 if (strcmp(WORD\_LIST[i], word) == 0) {  
 return 1;  
 }  
 }  
 return 0;  
}  
  
void test\_isReservedWord() {  
 assert(isReservedWord(3, "xor") == 1);  
 assert(isReservedWord(4, "case") == 1);  
 assert(isReservedWord(4, "else") == 1);  
 assert(isReservedWord(4, "bruh") == 0);  
 assert(isReservedWord(8, "register") == 1);  
}  
  
typedef enum State {  
 *START*,  
 *IN\_WORD*,  
 *TRASH*} State;  
  
int main() {  
 test\_isReservedWord();  
  
 char currentC;  
 char word[10] = {'-'};  
 int length = 0;  
 int max\_length = 10;  
 int res = 0;  
  
 State state = *START*;  
  
 while ((currentC = tolower(getchar())) != ']') {  
 switch (state) {  
 case *START*:  
 if (isspace(currentC)) {  
 length = 0;  
 break;  
 }  
  
 if (isalpha(currentC)) {  
 word[length] = currentC;  
  
 ++length;  
 state = *IN\_WORD*;  
 } else state = *TRASH*;  
  
 break;  
  
 case *TRASH*:  
 length = 0;  
  
 if (isspace(currentC))  
 state = *START*;  
  
 break;  
  
 case *IN\_WORD*:  
 if (isalpha(currentC)) {  
 if (length >= max\_length) {  
 length = 0;  
 state = *TRASH*;  
 }  
  
 word[length] = currentC;  
 ++length;  
 break;  
 }  
  
 if (isspace(currentC)) {  
 word[length] = '\0';  
 res += isReservedWord(length, word);  
 length = 0;  
 state = *START*;  
 break;  
 }  
  
 state = *TRASH*;  
 break;  
  
 default:  
 break;  
 }  
 }  
 printf("\n%d\n", res);  
  
 return 0;  
}

**9. Дневник отладки**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  | или |  |  |  |  |  |
|  | дом. |  |  |  |  |  |
| 1 | дом | 18.11.2022 – 24.11.2022 | Весь день | Ничего не работает | Пытался дебажить | Безуспешно |
| 2 | дом | 18.12.2022 | 11:00 | Текущее слово считывалось полностью, хотя я его ограничивал по размеру | В определённую ячейку массива чаров можно записать \0 чтобы обозначит конец слова | Прикольный трюк |

1. **Замечания автора**

Эта лаба не очень актуальна, потому что само задание вообще какое-то бессмысленное, ведь его надо решать как МТ, поэтому вся польза была в том, что мы увидели неприятные особенности языка Си и попытались их обойти

1. **Выводы**

Зачет Thumbs Up Emoji (U+1F44D). Когда я писал эту лабу в первый раз, тут был гневный комментарий, но теперь, после изменения всего кода, после того, как я набрался немного опыта в программировании на Си, эта лаба оказалась намного приятнее, хотя, конечно, ЛР #11 это, безусловно, худшая ЛР в первом семе.

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

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