

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

Студент группы М80-109Б-22 Шамбилов Руслан Талгатович, № 23 по списку

Контакты e-mail ruslanshmbiloff@yandex.ru,
telegram @willr4in.

Работа выполнена: «10» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим
Алексеевич

Отчет сдан « » декабрь 2022 г., итоговая оценка

Подпись преподавателя

1.Тема: Обработка последовательности литер входного текстового файла.

2. Цель работы: Научиться работать с конечными автоматами.

3. Задание: Ко всем допустимым числам добавить ведущий ноль.

4. Оборудование (студента):

Процессор AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz с ОП 16000Мб, НМД 512 Гб.
Монитор 1920x1080

5. Программное обеспечение (студента):

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *18.10 cosmic*
интерпретатор команд: *bash* версия *4.4.19*.
Система программирования -- версия --, редактор текстов *emacs* версия *25.2.2*

6. Идея, метод, алгоритм

Работать со switch, case. Понадобится 4 кейса, то есть 4 состояния, каждое из которых будет выполнять то или иное действие.

7. Сценарий выполнения работы

Входные данные	Выходные данные	Описание тестируемого случая
200 110 2 3 *	0200 0110 02 03	
501 31 3b 1 4a *	0501 031 01	

8. Распечатка протокола

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

```
#include <assert.h>
```

```
char ASCII(int symbol) {  
    if (symbol == 48)  
        return '0';  
    else if (symbol == 49)  
        return '1';  
    else if (symbol == 50)  
        return '2';  
    else if (symbol == 51)  
        return '3';  
    else if (symbol == 52)  
        return '4';  
    else if (symbol == 53)  
        return '5';  
    else if (symbol == 54)  
        return '6';  
    else if (symbol == 55)  
        return '7';  
    else if (symbol == 56)  
        return '8';  
    else if (symbol == 57)  
        return '9';  
}
```

```
void test_ASCII() {  
    assert(ASCII(48) == '0');  
    assert(ASCII(53) == '5');  
    assert(ASCII(49) == '1');  
    assert(ASCII(54) == '6');  
    assert(ASCII(50) == '2');  
}
```

```
int main() {  
    test_ASCII();  
  
    int state = 1, count = 0;  
    int symbol;  
  
    while((symbol = getchar()) != '*') {  
        char array[100];  
        switch (state) {
```

case 1:

```
if (symbol == ' ' || symbol == ';' || symbol == '\n' || symbol == '\t') {  
    state = 1;  
}  
else if ((symbol >= 'A' && symbol <= 'Z') || (symbol >= 'a' && symbol <= 'z')) {  
    state = 3;  
}  
else if (symbol >= '0' && symbol <= '9') {  
    array[count] = ASCII(symbol);  
    count += 1;  
    state = 2;  
}  
break;
```

case 2:

```
if (symbol >= '0' && symbol <= '9') {  
    array[count] = ASCII(symbol);  
    count += 1;  
    state = 2;  
}  
else if ((symbol >= 'A' && symbol <= 'Z') || (symbol >= 'a' && symbol <= 'z')) {  
    state = 3;  
}  
else if (symbol == ' ' || symbol == ';' || symbol == '\n' || symbol == '\t') {  
    printf("%c", '0');  
    for (int x = 0; x < count; x++) {  
        printf("%c", array[x]);  
    }  
    printf("%c", ' ');  
    count = 0;  
    state = 4;  
}  
break;
```

case 3:

```
if ((symbol >= '0' && symbol <= '9') || (symbol >= 'A' && symbol <= 'Z') || (symbol >= 'a'  
&& symbol <= 'z')) {  
    count = 0;  
    state = 3;  
}  
else if (symbol == ' ' || symbol == ';' || symbol == '\n' || symbol == '\t') {  
    count = 0;  
    state = 1;  
}  
break;
```

case 4:

```
if (symbol >= '0' && symbol <= '9') {
```

```

        array[count] = ASCII(symbol);
        count += 1;
        state = 2;
    }
    else if ((symbol >= 'A' && symbol <= 'Z') || (symbol >= 'a' && symbol <= 'z')) {
        state = 3;
    }
    else if (symbol == ' ' || symbol == ';' || symbol == '\n' || symbol == '\t') {
        state = 1;
    }
    default:
        break;
}
}
return 0;
}

```

9. Дневник отладки

№	Лаб.	Дата	Время	Событие	Действие по исправлению	Примечание
1	дом	9 декабря	19:56	Написал прогу для 3 кейсов	Надо переписать для 4, ибо не хватает состояния для мусора	
2	дом	11 декабря	21:20	Все хорошо работает, но меняет при вводе 1 на 0	Поправил некоторые кейсы, все работает	

10. Замечания автора

Нет.

11. Выводы

Работу с конечным автоматом оценил бы на 8/10, для меня одна из самых лучших и красивых (в плане кода) лабораторных. Сначала написание автомата и самого кода давались сложным, но в процессе изучения все стало понятно. Также хочется отметить конструкцию switch и case, которая лежит в основе программы.

Подпись студента
