

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

Студент группы М8О-109Б-22 Недосекин Александр  
Александрович, № по списку 13

Контакты vk, telegram @Nedosekin\_Aleksandr

Работа выполнена: «20» ноября 2022 г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « » 2022 г., итоговая оценка \_\_\_\_\_

Подпись преподавателя

---

1. **Тема:** Обработка последовательности литер входного текстового файла. Простейшие приёмы лексического анализа. Диаграммы состояний и переходов.
2. **Цель работы:** Научиться обрабатывать последовательности литер входного текстового текста.
3. **Задание (вариант 24):** Найти количество восьмеричных чисел в диапазоне от 10 до 1000
4. **Оборудование** (студента):  
Процессор *Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz* с ОП 5,88 Гб, НМД 1024 Гб. Монитор 1920x1080
5. **Программное обеспечение** (студента):  
Операционная система семейства: *linux*, наименование: *ubuntu*, версия 22.04.1 LTS Интерпретатор команд: *bash* версия 5.1.16.  
Система программирования -- версия --, редактор текстов *emacs* версия 28.2 Утилиты операционной системы --  
Прикладные системы и программы --  
Местонахождение и имена файлов программ и данных на домашнем компьютере /home/tng00
6. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)
7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

1. Изучить материалы лекции
2. Составить алгоритм действий, переходов состояний
3. Написать программу, удовлетворяющую условиям лабораторной работы 4. Провести unit-тестирование
5. Запротоколировать сеанс

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

// В зависимости от встреченного символа выбирается режим программы.

//Если встречается число из 2 восьмеричных цифр, то переменная *amount*, содержащая в себе кол-во чисел увеличивается на один.

//Если встречается третья восьм. Цифра, то *amount* остаётся прежним.

//Если встречается четвёртая цифра подряд (кроме случая с числом 1000), то значение *amount* откатывается до предыдущего значения.

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

```

#include <ctype.h>
//проверка на восьмеричное
int atype(char a)
{

    if (a>='0' && a<='7') return 1;
    else return 2;

};

int main(){

    int state = 1;
    int symbol;
    int amount = 0;
    int amount2 = 0;

    while ((symbol = fgetc(stdin)) != EOF) {

        switch (state) {

            case 1:
                amount2 = amount;
                if (atype(symbol) == 1) state = 2;

                else
                    {state = 1;
                     break;}

            case 2:
                if (atype(symbol) == 1) state = 3;
                else state = 1;
                break;

            case 3:
                if (atype(symbol) == 1) {state = 4;
                    amount = amount + 1;}
                else
                    state = 1;

                break;

            case 4:
                if (atype(symbol) == 1)
                    {state = 5;
                     break;}

                if ((symbol == '8') || (symbol == '9')) {
                    state = 1;
                    amount = amount2;
                    break;

                if (atype(symbol) == 2) state = 1;

```

```

        break;
    }

    case 5:
    if (symbol == '0') {
        state = 1;
        break;
    }
    if ((atype(symbol) == 1) || (symbol == '8') || (symbol == '9')) {
        state = 5;
        amount = amount2;
        break;
    }

    if (atype(symbol) == 2) state = 1;
    break;

    }
}
printf("%d\n", amount );
return 0;
}

```

#### **10. Замечания автора по существу работы**

#### **11. Выводы**

Выяснил, что из себя представляет конечный автомат. Научился обрабатывать последовательность литер, используя язык Си, познакомился с диаграммами состояний и переходов, их использованием для решения задач.

Подпись студента