

Отчет по лабораторной работе № 1¹ по курсу _____

Студент группы 108 Касирин Кирилл, № по списку 8

Контакты www, e-mail, icq, skype _____

Работа выполнена: « 02 ноября 2020 г.

Преподаватель: _____ каф. 806 _____

Входной контроль знаний с оценкой _____

Отчет сдан « » 201 г., итоговая оценка _____

Подпись преподавателя _____

1. Тема: Обработка последовательности литер входного текстового файла. Простейшие приёмы лексико-грамматического анализа. Диаграммы состояний и переходов.

2. Цель работы: Составить программы на языке Си, выполняющую анализ и обработку вводимого текста в соответствии с условием задания.

3. Задание (вариант №32): Выделить все десятичные числа от 17 до 77 по модулю и распечатать их значения в словесной форме по-русски.

4. Оборудование (лабораторное):
ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____.
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:
Процессор _____ с ОП _____ Мб, НМД _____ Мб. Монитор _____
Другие устройства _____

5. Программное обеспечение (лабораторное):
Операционная система семейства _____, наименование _____ версия _____
интерпретатор команд _____ версия _____
Система программирования _____ версия _____
Редактор текстов _____ версия _____
Утилиты операционной системы _____

Прикладные системы и программы _____
Местонахождение и имена файлов программ и данных _____

Программное обеспечение ЭВМ студента, если использовалось:
Операционная система семейства _____, наименование _____ версия _____
интерпретатор команд _____ версия _____.
Система программирования _____ версия _____
Редактор текстов _____ версия _____
Утилиты операционной системы _____

Прикладные системы и программы _____
Местонахождение и имена файлов программ и данных на домашнем компьютере _____

6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Назначаем состояния, которые определяют начало последовательности, первая на вход цифра у числа, вторая на вход цифра у числа, конец числа(если число не двузначное) и букву или иной символ.

Запоминаем первую и вторую цифру числа.Проверяем двузначное ли число(для этого считываем следующий символ после двух цифр).

Проверяем условие: число от 17 до 77.Если число удовлетворяет условиям,с помощью функции выводим число в словесной форме по-русски,

Иначе выводим эти цифры, а так же числа, не удовлетворяющие условиям, или иные символы, не изменяя.

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

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя _____

8. Распечатка протокола (подклейте листинг окончательного варианта программы с тестовыми примерами, подписаный преподавателем).

main.c

```
#include <stdio.h>
#include <stdbool.h>
#include "library.h"

int main(){
    int Size = 50;
    char buffer[Size];
    enum State currentState = Start;
    unsigned count = 0;
    unsigned digit = -2;
    char fdigit;
    char sdigit;
    char pdigit;
    while (fgets(buffer,Size,stdin)!= NULL){
        for (int i=0;i< Size && buffer[i]!='\0';++i){
            currentState = prog(currentState, buffer[i]);
            if (currentState == FirstDigit){
                fdigit = 'a';
                sdigit = 'a';
                fdigit = buffer[i];
                count += 1;
                digit = -1;
            }
            if (currentState == SecondDigit){

                digit=0;
                bool condition = (fdigit == '1' && buffer[i]>='7')||(fdigit == '7' && buffer[i] <= '7')||((fdigit > '1') && (fdigit < '7'));
                if (condition){
                    sdigit=buffer[i];
                    count+=1;
                    digit=1;
                }
                else{
                    printf("%c%c",fdigit,buffer[i]);
                }
                pdigit = buffer[i];
            }
            if (currentState == Letter){

                if (count == 2){

                    conclusion(fdigit,sdigit);
                    digit = 0;
                }
                if (digit == -1){

                    printf("%c",fdigit);
                    digit=0;
                }
                printf("%c",buffer[i]);
                count = 0;
            }
            if (currentState == EndUnNum){

                count = 0;
                if (digit == 1){

                    printf("%c%c",fdigit,pdigit);
                }
                printf("%c",buffer[i]);
            }
        }
    }
}
```

library.h

```
enum State {Start=0,FirstDigit,SecondDigit,EndUnNum,Letter};
enum State prog(enum State currentState,char currentSymbol);
void conclusion(char c,char d);
```

functions.c

```
#include <stdio.h>
#include <stdbool.h>
#include "library.h"

enum State prog(enum State currentState,char currentSymbol)
{
    enum State newState;
    switch(currentState){
        case Start:
        {
            if (currentSymbol > '0' && currentSymbol < '8'){
                newState = FirstDigit;
            } else newState = Letter;
            break;
        }
        case Letter:
        {
            if (currentSymbol > '0' && currentSymbol < '8'){
                newState = FirstDigit;
            } else newState = Letter;
            break;
        }
        case FirstDigit:
        {
            if (currentSymbol >='0' && currentSymbol <='9'){
                newState = SecondDigit;
            } else newState = Letter;
            break;
        }
        case SecondDigit:
        {
            if (currentSymbol >= '0' && currentSymbol <= '9'){
                newState = EndUnNum;
            } else newState = Letter;
            break;
        }
        case EndUnNum:
        {
            if (currentSymbol >= '0' && currentSymbol <= '9'){
                newState = EndUnNum;
            } else newState = Letter;
            break;
        }
    }
    return newState;
}
void printer(char c,char d){
    if (c=='1' && d=='7'){
        printf("семнадцать");
    }
    else if (c=='1'& d=='8'){
        printf("восемнадцать");
    }
    else if (c=='1'& d=='9'){
        printf("девятнадцать");
    }
    else {
        if (c=='2') printf("двадцать");
        if (c=='3') printf("тридцать");
        if (c=='4') printf("сорок");
        if (c=='5') printf("пятьдесят");
        if (c=='6') printf("шестьдесят");
        if (c=='7') printf("семьдесят");
        if (d=='1') printf(" один");
        if (d=='2') printf(" два");
        if (d=='3') printf(" три");
        if (d=='4') printf(" четыре");
        if (d=='5') printf(" пять");
        if (d=='6') printf(" шесть");
        if (d=='7') printf(" семь");
        if (d=='8') printf(" восемь");
        if (d=='9') printf(" девять");
    }
}
```

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. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание

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

11. Выводы

Я составил программу на языке Си, выполняющую анализ и обработку вводимого текста в соответствии с условием задания.

Недочёты при выполнении задания могут быть устранены следующим образом: _____

Подпись студента _____