

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В. И. УЛЬЯНОВА (ЛЕНИНА)**  
**КАФЕДРА МО ЭВМ**

**ОТЧЕТ**  
**по лабораторной работе №3**  
**по дисциплине «Программирование»**  
**Тема: Обход файловой системы**

Студент гр. 0382

Афанасьев Н. С.

Преподаватели

Берленко Т. А.

Санкт-Петербург

2021

## Цель работы.

Изучение работы с файловой системой на языке C.

## Задание.

Задана иерархия папок и файлов по следующим правилам:

- Название папок может быть только "add" или "mul"
- В папках могут находиться другие вложенные папки и/или текстовые файлы
- Текстовые файлы имеют произвольное имя с расширением .txt
- Содержимое текстовых файлов представляет собой строку, в которой через пробел записано некоторое количество целых чисел

Требуется написать программу, которая, запускается в корневой директории, содержащей одну папку с именем "add" или "mul" и вычисляет и выводит на экран результат выражения состоящего из чисел в поддиректориях по следующим правилам:

- Если в папке находится один или несколько текстовых файлов, то математическая операция определяемая названием папки (add = сложение, mul = умножение) применяется ко всем числам всех файлов в этой папке
- Если в папке находится еще одна или несколько папок, то сначала вычисляются значения выражений, определяемые ими, а после используются уже эти значения

## Выполнение работы.

В функции `int main()` с помощью функции `fopen` библиотеки `stdlib` открывается файл `result.txt`, куда будет записываться ответ. С помощью функции `fprintf` записывается результат рассматриваемой ниже функции `calculate`. Далее, с помощью функции `fclose` файл закрывается.

Функция *int calculate(char\* path, char\* dir\_name)* является рекурсивной функцией, выполняющую подсчёт значений в файлах в определённой папке. Функция принимает текущий путь и имя папки, которую следует рассмотреть. В начале формируется строка, содержащая путь, включая рассматриваемую папку. Далее, с помощью функции *opendir* библиотеки *dirent.h* эта папка открывается и записывается в качестве объекта структуры *DIR*. В цикле *while* рассматривается содержимое папки. Если рассматриваемый элемент – файл (тип элемента определяется по значению поля *d\_type*), то сначала формируется строка, содержащая путь к файлу, а затем этот файл открывается и считывается с помощью *fopen* и *fscanf* соответственно. В зависимости от названия родительской папки, значения в файлах либо складываются, либо умножаются. После, файл закрывается и рассматривается следующий элемент. Если элемент представляет из себя папку, то, в зависимости от операции, прибавляется или умножается значение функции *calculate* для рассматриваемой папки – создаётся рекурсия. В итоге, функция закрывает папку с помощью функции *closedir* и возвращает значение чисел в файлах.

Разработанный программный код см. в приложении А.

## Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	Root Add Add file.txt: 1 file1.txt: 1 Mul file2.txt: 2 2 file3.txt: 7 Add file4.txt: 1 2 3 file5.txt: 3 -1	226	Верно

## Выводы.

Была изучена работа с файлами на языке C.

Разработана программа, которая совершает обход файловой системы, в зависимости от названия папок - “add” или “mul” - рассчитывает значение чисел в файлах внутри вложенных папок, а затем записывает получившийся результат в текстовый документ.

## ПРИЛОЖЕНИЕ А

### ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.c

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <dirent.h>
#include <string.h>

int calculate(char* path, char* dir_name){
    char* newPath;
    if(!strcmp(dir_name, "")) newPath = path;
    else{
        newPath = malloc((strlen(path) + strlen(dir_name) +
10)*sizeof(char));
        sprintf(newPath, "%s/%s", path, dir_name);
    }

    DIR* D = opendir(newPath);
    struct dirent* dir;

    int result = strcmp(dir_name, "add") ? 1 : 0;

    while((dir = readdir(D)) != NULL){
        if(dir->d_type & DT_DIR){
            if(strcmp(dir->d_name, ".") && strcmp(dir->d_name, "..")){
                if(!strcmp(dir_name, "add")) result +=
calculate(newPath, dir->d_name);
                else result *= calculate(newPath, dir->d_name);
            }
        }
        else{
            char* filePath = malloc((strlen(newPath) + strlen(dir-
>d_name) + 10)*sizeof(char));
            sprintf(filePath, "%s/%s", newPath, dir->d_name);
            FILE* stream = fopen(filePath, "r");

            int i ;
            while(!feof(stream)){
                fscanf(stream, "%d", &i);
                if(!strcmp(dir_name, "add")) result += i;
                else result *= i;
            }
            free(filePath);
            fclose(stream);
        }
    }
    if(strcmp(dir_name, "")) free(newPath);
    closedir(D);
    return result;
}

int main() {
    FILE* result = fopen("./result.txt", "w");
    fprintf(result, "%ld", calculate("./tmp", ""));
}
```

```
    fclose(result);  
    return 0;  
}
```