**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по лабораторной работе №3**

**по дисциплине «Программирование»**

ТЕМА: **Строки. Рекурсия, циклы, обход дерева**.

|  |  |  |
| --- | --- | --- |
| Студент гр. 6303 |  | Куликов М.Л. |
| Преподаватель |  | Берленко Т. А. |

Санкт-Петербург

2017

**Цель работы:**

Приобретение навыков работы с деревом директорий и файлов и считывания содержимого файлов.

**Задание:**

Дана некоторая корневая директория, в которой может находиться некоторое количество папок, в том числе вложенных. В этих папках хранятся некоторые текстовые файлы, имеющие имя вида **<filename>.txt**. В качестве имени файла используется символ латинского алфавита.

На вход программе подается строка. Требуется найти и вывести последовательность полных путей файлов, имена которых образуют эту строку.

**Ход работы:**

1. Подключены необходимые библиотеки и объявлены константы

//Подключение библиотек

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/types.h>

#include <dirent.h>

//Определение констант

#define DIR\_SIZE 100

#define STR\_SIZE 100

#define PATH\_SIZE 500

1. Объявление структуры данных, в которой хранятся символ строки, путь и флаг для проверки наличия символа в строке.

a typedef struct letter {

char character;

char path[PATH\_SIZE];

int flag;

} letter;

1. Описание функции, которая проверяет наличие имени файла (1 символ) в введенной строке.

void letter\_locating\_check(letter \*letter\_path, char \*current\_dir, char \*file\_name, int n) {

for (int i = 0; i < n; ++i) {

if (((letter\_path + i)->symbol == \*file\_name) && !((letter\_path + i)->flag)) {

\*(file\_name + strlen(file\_name)) = '.';

int current\_dir\_len = strlen(current\_dir);

strcat(current\_dir, "/");

strcpy((letter\_path + i)->path, strcat(current\_dir, file\_name));

(letter\_path + i)->flag = 1;

\*(current\_dir + current\_dir\_len) = '\0';

break;

}

}

}

1. Описание функции для проверки наличия всех символов введенной строки среди имен файлов дирктории.

int str\_collect\_check(letter \*letter\_path, int length) {

for (int i = 0; i < length; ++i)

if (!((letter\_path + i)->flag)) {

return 0;

}

return 1;

}

1. Описание функции для рекурсивного прохода по директории.

int dir\_traveller (char \*dir\_name, letter \*letter\_path, int n) {

char current\_dir[DIR\_SIZE];

strcpy(current\_dir, dir\_name);

DIR \*dir = opendir(current\_dir);

struct dirent \*de = readdir(dir);

if(dir) // проверка существования директории

while (de) {

if (de->d\_type == DT\_REG) {

char \*file\_name = strtok(de->d\_name, ".");

letter\_locating\_check(letter\_path, current\_dir, file\_name, n);

if (str\_collect\_check(letter\_path, n))

return 0;

}

if ((de->d\_type == DT\_DIR) && strcmp(".", de->d\_name) && strcmp(de->d\_name, "..")) {

int current\_dir\_len = strlen(current\_dir);

strcat(current\_dir, "/");

strcat(current\_dir, de->d\_name);

dir\_traveller(current\_dir, letter\_path, n);

\*(current\_dir + current\_dir\_len) = '\0';

}

de = readdir(dir);

}

closedir(dir);

return 0;

}

1. Код основной части.

int main () {

char dir\_name[DIR\_SIZE] = {'.', '\0'};

char str[STR\_SIZE];

fgets(str, (STR\_SIZE + 2), stdin);

int length = (strlen(str) - 1);

letter \*letter\_path = (letter\*)malloc(length \* sizeof(letter));

for (int i = 0; i < length; ++i) {

(letter\_path + i)->symbol = \*(str + i);

(letter\_path + i)->flag = 0;

}

dir\_traveller(dir\_name, letter\_path, length);

for (int i = 0; i < length; ++i) {

printf("%s\n", (letter\_path + i)->path);

}

free(letter\_path);

return 0;

}

1. Затем код работы вместе с отчетом были загружены на github:

* Создана новая ветка с помощью команды *git checkout –b kulikov\_sem2\_lab3*
* Добавлены файлы для загрузки с помощью команды *git add*
* Добавлен комментарий коммита командой *git commit –m “Lab3 ”*
* Файлы были загружены командой *git push origin kulikov\_sem2\_lab3*

**Вывод:** В ходе лабораторной работы были получены навыки работы с деревом директорий и файлов.