**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: Рекурсия, циклы, обход дерева

|  |  |  |
| --- | --- | --- |
| Студент гр. 6303 |  | Ваганов Н.А. |
| Преподаватель |  | Берленко Т.А. |

Санкт-Петербург

2017

**Цель работы:**

**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 для работы заголовочные файлы

#include <sys/types.h>

#include <stdlib.h>

#include <stdio.h>

#include <string.h>

#include <dirent.h>

1. Создана функция nextDir для рекурсивного обхода директорий. Функция на вход получает искомый символ, который ищет в заданных директориях. При нахождении нужного символа путь запоминается.

int nextDir (char \*\* banList, char \* way, char letter, int \* sizeOfWay, int \* sizeOfBanList)

{

int wayLen = strlen(way);

DIR \*dir;

struct dirent \*entry;

dir = opendir(way);

while (entry = readdir(dir))

{

if(strstr(entry->d\_name, ".txt"))

{

char \* buf = &entry->d\_name[strlen(entry->d\_name)-4];

if(!strcmp(buf, ".txt"))

if(letter == entry->d\_name[0])

{

strcat(way, entry->d\_name);

int i;

int flag = 0;

for(i = 0; i<(\*sizeOfBanList); i++)

{

if(banList[i])

if(!strcmp(banList[i], way))

{

flag = 1;

break;

}

}

if(!flag)

{

for(i = 0; i<(\*sizeOfBanList); i++)

{

if(banList[i] == NULL)

{

banList[i] = (char\*)malloc(sizeof(char)\*strlen(way));

strcpy(banList[i], way);

printf("%s\n", way);

closedir(dir);

return 1;

}

}

}

way[wayLen]='\0';

}

}

if(!strchr(entry->d\_name, '.'))

{

strcat(way, entry->d\_name);

strcat(way, "/");

if(nextDir(banList, way, letter, sizeOfWay, sizeOfBanList))

{

closedir(dir);

return 1;

}

way[wayLen] = '\0';

}

}

closedir(dir);

return 0;

}

1. В функции main() производится инициализация необходимых переменных, а также выделение памяти. Функция получает на вход строку, после чего к каждому символу строки применяется функция nextDir().

int main()

{

int sizeOfWay = 100;

int sizeOfBanList = 20;

int i;

char inputStr[100];

char \* way = (char\*)malloc(sizeof(char)\*sizeOfWay);

char \*\* banList = (char\*\*)malloc(sizeof(char\*)\*sizeOfBanList);

for(i = 0; i<sizeOfBanList; i++)

banList[i] = NULL;

strcat(way, "./");

printf("Введите строку: \n");

fgets(inputStr, 99, stdin);

for (i=0;i<strlen(inputStr)-1;i++)

{

if(!nextDir(banList, way, inputStr[i], &sizeOfWay, &sizeOfBanList))

printf("Файла с названием '%c' не существует\n", inputStr[i]);

way[2] = '\0';

}

free(way);

free(banList);

return 0;

}

1. Затем файл main.c с кодом написанной программы и отчёт о проведенной работе был загружен на github c помощью консоли:

* Создана новая ветка vaganov\_sem2\_lab3

git checkout –b vaganov\_sem2\_lab3

* Создана новая директория vaganov\_sem2\_lab3 и файл main.c в ней, куда помещен код написанной программы

mkdir vaganov\_sem2\_lab3

cd vaganov\_sem2\_lab3

nano main.c

* Созданные файлы добавлены для загрузки

git add vaganov\_sem2\_lab3

* Добавлен комментарий

git commit –m lab3sem2

* Файлы загружены на репозиторий Git

git push origin vaganov\_sem2\_lab3

**Вывод:** В ходе работы создана функция для рекурсивного обхода директорий и поиска файла с заданным именем. Для выполнения поставленной задачи были использованы циклы, рекурсия, а также динамическая память.