**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: Рекурсивный обход директорий

|  |  |  |
| --- | --- | --- |
| Студентка гр. 6304 |  | Иванкова В.М. |
| Преподаватель |  | Берленко Т.А. |

Санкт-Петербург

2017

**Оглавление**

[**Цель работы:** 3](#_Toc484123229)

[**Задание:** 3](#_Toc484123230)

[**Содержание:** 3](#_Toc484123231)

[**Вывод** 5](#_Toc484123232)

[**Приложение** 5](#_Toc484123233)

# **Цель работы:**

Написать программу, которая находит и выводит последовательность полных путей файлов, имена которых образуют заданную пользователем строку.

# **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.

# **Содержание:**

Для решения данной задачи была реализована рекурсивная функция checkDr, которая принимает на вход указатель на имя директории, указатель на путь директории и ее длину.

int checkDr(char \*nameDir, character \*chPath, int len) //проверка директории

{

char currentDir[100];//создаем строку, для хранения имени папки

strcpy(currentDir, nameDir);//скопировали имя директории в новый массив

DIR \*dir = opendir(currentDir);//создаем переменную типа DIR и открываем текущую папку

struct dirent \*content = readdir(dir);//создаём структуру типа dirent

if(dir)//если папка открыта

while(content)//в папке есть содержимое

{

if (content->d\_type == 8)//если это файл

{

char \*fileName = strtok(content->d\_name, ".");//имя файла до точки

checkCh(chPath, currentDir, fileName, len);//проверяем первую букву

if (isNameFull(chPath, len))//если имя полное, выходим

return 0;

}

if ((content->d\_type == 4) && strcmp(".", content->d\_name) && strcmp("..", content->d\_name))//если это директория

{

int currentDirLen = strlen(currentDir);//длина этой директории

strcat(currentDir, "/");//добавили файловый разделитель

strcat(currentDir, content->d\_name);//добавили имя файла

checkDr(currentDir, chPath, len);//проверки директории

\*(currentDir + currentDirLen) = '\0';//ставим конец строки

}

content = readdir(dir);//считываем содержимое из текущей папки

}

closedir(dir);//закрываем папку

return 0;

}

Так же, для выполнения данной задачи, были реализованы некоторые функции.

Функция сheckСh проверяет, начинается ли имя файла с заданной в строке буквы в данной директории.

void checkCh(character \*chPath, char \*currentDir, char \*fileName, int len) //функция ищет букву в имени файла

{

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

{

if (((chPath + i)->symbol == \*fileName) && !((chPath + i)->flag))//если текущий символ совпадает с именем файла

{

\*(fileName + strlen(fileName)) = '.';//поставить после имени файла точку

int currentDirLen = strlen(currentDir);//длина имени текущей папки

strcat(currentDir, "/");//добавляем к строке символ

strcpy((chPath + i)->path, strcat(currentDir, fileName));//копирует в текущий путь к файлу путь со слэшем

(chPath + i)->flag = 1;//активируем флаг

\*(currentDir + currentDirLen) = '\0';//добавляем символ концастроки

break;

}

}

}

Функция isNameFull проверяет, готово ли «слово» из букв заданной строки.

int isNameFull (character \*chPath, int len) //функция для проверки готовности слова

{

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

if (!(chPath + i)->flag)//если флаг не активен

return 0;

return 1;

}

Для реализации данной программы была введена специальная структура character.

typedef struct character //структура для хранения:

{

char symbol; //буквы

char path[Pathlength]; //директории

int flag; //флажок есть-нет

} character;

# 

# **Вывод**

В ходе данной лабораторной работы была создана программа, на основе алгоритма обхода директорий с помощью рекурсивной функции.

# **Приложение**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <sys/types.h> //различные типы данных

#include <dirent.h> //Открытие и вывод каталогов

#define Pathlength 1000

typedef struct character //структура для хранения:

{

char symbol; //буквы

char path[Pathlength]; //директории

int flag; //флажок есть-нет

} character;

int isNameFull (character \*chPath, int len) //функция для проверки готовности слова

{

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

if (!(chPath + i)->flag)//если флаг не активен

return 0;

return 1;

}

void checkCh(character \*chPath, char \*currentDir, char \*fileName, int len) //функция ищет букву в имени файла

{

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

{

if (((chPath + i)->symbol == \*fileName) && !((chPath + i)->flag))//если текущий символ совпадает с именем файла

{

\*(fileName + strlen(fileName)) = '.';//поставить после имени файла точку

int currentDirLen = strlen(currentDir);//длина имени текущей папки

strcat(currentDir, "/");//добавляем к строке символ

strcpy((chPath + i)->path, strcat(currentDir, fileName));//копирует в текущий путь к файлу путь со слэшем

(chPath + i)->flag = 1;//активируем флаг

\*(currentDir + currentDirLen) = '\0';//добавляем символ концастроки

break;

}

}

}

int checkDr(char \*nameDir, character \*chPath, int len) //проверка директории

{

char currentDir[100];//создаем строку, для хранения имени папки

strcpy(currentDir, nameDir);//скопировали имя директории в новый массив

DIR \*dir = opendir(currentDir);//создаем переменную типа DIR и открываем текущую папку

struct dirent \*content = readdir(dir);//создаём структуру типа dirent

if(dir)//если папка открыта

while(content)//в папке есть содержимое

{

if (content->d\_type == 8)//если это файл

{

char \*fileName = strtok(content->d\_name, ".");//имя файла до точки

checkCh(chPath, currentDir, fileName, len);//проверяем первую букву

if (isNameFull(chPath, len))//если имя полное, выходим

return 0;

}

if ((content->d\_type == 4) && strcmp(".", content->d\_name) && strcmp("..", content->d\_name))//если это директория

{

int currentDirLen = strlen(currentDir);//длина этой директории

strcat(currentDir, "/");//добавили файловый разделитель

strcat(currentDir, content->d\_name);//добавили имя файла

checkDr(currentDir, chPath, len);//проверки директории

\*(currentDir + currentDirLen) = '\0';//ставим конец строки

}

content = readdir(dir);//считываем содержимое из текущей папки

}

closedir(dir);//закрываем папку

return 0;

}

int main ()

{

char nameDir[100] = {'.', '\0'};//создаем строку, куда будем записывать путь

char s[100];//строка для хранения букв

fgets(s, (102), stdin); //читаем строку, в которой записана последовательность букв

int len = (strlen(s) - 1);//определяем длину строки

character \*chPath = (character\*)malloc(len \* sizeof(character));//создаем структуру типа character

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

{

(chPath + i)->symbol = s[i];//записываем каждую букву введенного слова в отдельную структуру

(chPath + i)->flag = 0;//флагу присваиваем значение 0

}

checkDr(nameDir, chPath, len);

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

printf("%s\n", (chPath + i)->path);//выводим путь, который искали

return 0;

}