Министерство образования и науки Российской Федерации

Новосибирский государственный технический университет

Кафедра теоретической и прикладной информатики

Лабораторная работа №1 по дисциплине

«Управление ресурсами в вычислительных системах»

Факультет: ПМИ

Группа: ПМИ-12

Студенты: Швадченко А. В.

Михайловский М. А.

Вариант: 4

Преподаватели: Стасышин В. М., Сивак М. А.

Новосибирск

2024

1. ***Цель работы***

Ознакомиться с устройством файловой системы ОС UNIX, механизмами ее функционирования, программными средствами для работы с ней (командный язык Shell, язык Си).

1. ***Задание***

Программа просматривает текущий каталог и выводит на экран имена всех встретившихся в нем файлов с заданным расширением (параметр программы). Затем осуществляется переход в родительский каталог, который затем становится текущим, и указанные выше действия повторяются до тех пор, пока текущим каталогом не станет корневой каталог.

1. ***Описание используемых структур***

**DIR \*opendir(char \*dirname)**

Открывает поток каталога и возвращает указатель на структуру типа DIR, которая содержит информацию о каталоге.

**struct dirent \*readdir(DIR \*ptr)**

Возвращает название следующего файла в каталоге. Иными словами, функция

readdir() читает оглавление каталога по одному файлу за раз. Параметр ptr должен

указывать на поток каталога, открытый с помощью opendir().

**int closedir(DIR \*ptr)**

Закрывает поток каталога, на который указывает ptr. Возвращает 1 в случае успеха и

0 в противном случае. При неудаче переменная errno устанавливается равной EBADF

(недействительный каталог).

**char \*getcwd(char \*buf, size\_t size)**

Копирует абсолютный путь к текущему рабочему каталогу в массиве, на который

указывает buf, имеющий длину size. Если текущий абсолютный путь требует буфера,

длина которого превышает size, то возвращается NULL, а errno принимает значение

ERANGE.

**int chdir(const char \*path)**

Изменяет текущий каталог каталог на path. В случае успеха возвращается ноль, в

противном случае возвращается -1.

1. ***Описание алгоритма на языке Shell***
2. Проверяем, указан ли аргумент при вызове скрипта
3. Выводим название каталога
4. Выводим содержимые файлы каталога с нужным расширением: при помощи команды ls получаем список (-1) файлов, оканчивающиеся нужным расширением.
5. Переходим в родительский каталог
6. Повторяем п.п. 2-4 до тех пор, пока результат команды pwd не станет равны строке “/”
7. Переходим в корневой каталог и аналогично выводим его содержимое
8. ***Описание алгоритма на языке C***
9. Передаем строку (расширение) в качестве аргумента программы и

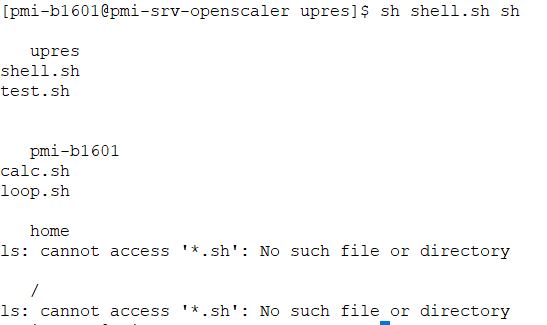
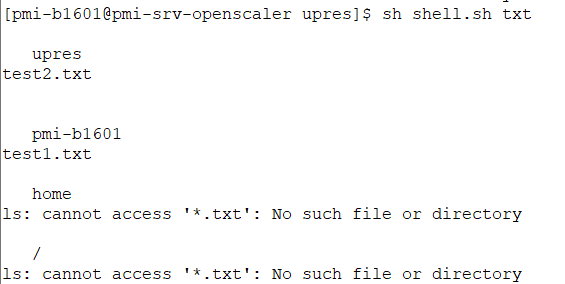
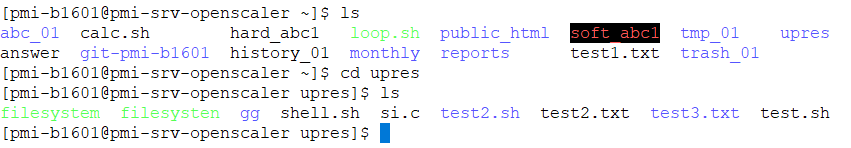
проверяем, чтобы argc был равен 1

1. Открываем текущий каталог
2. Получаем и выводим путь текущего каталога
3. Получаем и выводим на экран файлы с нужным расширением
4. Переходим в родительский каталог и закрываем текущий
5. Повторяем п.п. 3-5 до тех пор, пока путь каталога отличается от “/”
6. ***Спецификация***

Программы находятся в каталоге /home/pmi-b1601/upres и называются shell.sh и si.c/.

Чтобы запустить скрипт bash необходимо выполнить команду: sh shell.sh [расширение], где [расширение] – тип расширения, нужный для нахождения файлов, например **txt**. В результате получим список каталогов и содержащихся в них файлов с заданным расширением.

Чтобы получить исполняемый файл необходимо выполнить команду: gcc –o [имя\_исполняемого\_файла] si.c. Далее можно запустить исполняемый файл командой: ./[ имя\_исполняемого\_файла] [расширение]. В результате получим список каталогов и содержащихся в них файлов с заданным расширением.

1. ***Тесты на языке Shell***
2. Команда: sh shell.sh sh

Выведены корректно файлы с этим расширением.

1. Команда: sh shell.sh txt

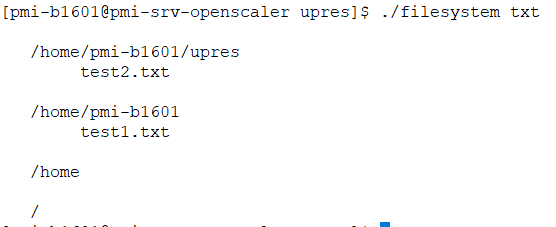
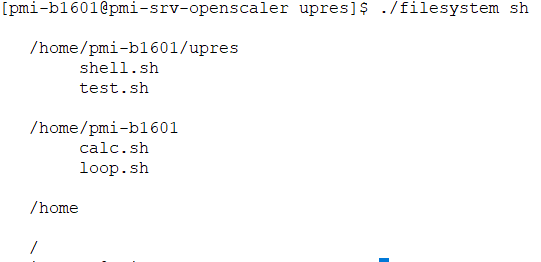
Выведены корректно файлы с этим расширением.

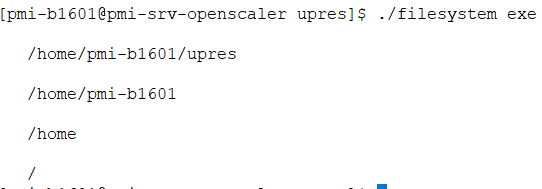
1. Команда: sh shell.sh exe

Файлов с таким расширением не найдены

1. Команда: sh shell.sh

Не указан параметр программы.

1. *********Тесты на языке С***



Тесты аналогичны тестам на языке Shell.

1. ***Текст программы на языке Shell***

#!/bin/bash

if [ ! $# -eq 1 ]

then

echo "Error: check the input data." # проверка количества переданных параметров

exit 2

fi

until [ "`pwd`" = "/" ] # пока не дошли до корневого каталога

do

CURRENT=`pwd` # текущий путь

echo -e "\n $(basename "$CURRENT")" # форматирование и вывод названия текущего каталога

ls -p \*.$1 | grep -v : # вывод имен файлов с нужным расширением

cd .. # переход в корневой каталог

done

cd .. # переход в родительский каталог

echo -e "\n /"

ls -p \*.$1 # вывод имен файлов с нужным расширением

1. ***Текст программы на языке C***

#include <unistd.h>

#include <dirent.h>

#include <stdio.h>

#include <errno.h>

#include <string.h>

#include <stdlib.h>

#include <sys/types.h>

#include <sys/stat.h>

int main(int argc, char \*argv[])

{

DIR \*d; //указатель на дескриптор каталога

char cwd[512]; // буфер, содержащий путь каталога

char \*e =".";

char \*ex = argv[1];

char \*exten = (char \*) malloc(1+strlen(e)+strlen(ex));

strcpy(exten, e);

strcat(exten, ex);

printf("%s", exten);

struct dirent \*ent;

struct stat stat\_buff;

if (argc == 1)// проверка на наличие аргумента

{

perror("Error: Check the input data!\n");

exit(EXIT\_FAILURE);

}

do

{

d = opendir("."); // открытие начального каталога

if (getcwd(cwd, sizeof(cwd)) != NULL)

fprintf(stdout, "\n %s\n", cwd);// запись и вывод пути каталога

if (d)

{

while ((ent = readdir(d)) != NULL)

if ((lstat(ent->d\_name, &stat\_buff))==0)

{

if(!(stat\_buff.st\_mode & S\_IFDIR))

{

if (strstr(ent->d\_name, exten) != NULL && ent->d\_name[0]!='.')

printf(" %s\n", ent->d\_name);// вывод имен файлов с нужным расширением

}

}

chdir(".."); // переход в родительский каталог

closedir(d); // закрытие каталога

}

} while ((int)cwd[1] != 0); // пока не дойдет до корневого каталога

return 0;

}