Министерство образования и науки Российской Федерации

Новосибирский государственный технический университет

Кафедра программных баз и систем данных

Лабораторная работа №2 по дисциплине

«Управление ресурсами в вычислительных системах»

Факультет: ПМИ

Группа: ПМ-04

Студентки: Анисимова К.Н.

Сулейманова К.А.

Вариант: 4

Преподаватель: Быханов К.В.

Новосибирск

2013

1. ***Цель работы***

Ознакомиться с файловой системой ОС UNIX, механизмами ее функционирования, основными элементами файловой системы: суперблок, описатели файлов, типы файлов, список свободных описателей файлов, список свободных блоков.

1. ***Задание***

Разработать программу, которая выводит на экран имена тех каталогов, которые находятся в текущем каталоге и не содержат в себе подкаталогов.

1. ***Решение***

Программа корректно работает, если в текущем каталоге есть каталоги, у которых нет разрешения на выполнение или доступ. В этом случае, мы убираем их рассмотрения и выводим сообщение об ошибке. Если открытие начального каталога невозможно, тоже выводится сообщение об этом.

1. ***Используемые команды и структуры***

struct stat, а именно поле st\_mode, в котором мы при помощи макроса S\_ISDIR проверяем флаг S\_IFDIR – каталог. Если S\_ISDIR в качестве передаваемого значения передано поле, соответствующее каталогу, он вернёт 1, иначе 0.

struct dirent, а именно поле d\_name и неявно поле d\_ino (при чтении каталога)

Системный вызов DIR \*opendir (char \*dirname), возвращающий указатель на дескриптор каталога или NULL при ошибке открытия.

Системный вызов int closedir (DIR \*dirptr), закрывающий открытый каталог.

Системный вызов int chdir (char \*pathname) для смены текущего каталога (в нашем случае смена текущего каталога необходима для вызова stat).

Системный вызов struct dirent \*readdir (DIR \*dirptr) для чтения записей каталога (возвращает NULL при ошибках чтения содержимого каталога).

Системный вызов int stat(const char \*name, struct stat \*stbuf) для получения информации о файле.

1. ***Текст программы***

#include <sys/types.h>

#include <sys/stat.h>

#include <unistd.h>

#include <dirent.h>

#include <stdio.h>

int main()

{

int flag;

DIR \*track;

DIR \*subdir;

struct dirent \*dir1, \*dir2;

struct stat cur\_file;

track = opendir(".");//текущий каталог

if(!track) //не открывается

{

printf("ошибка при открытии текущего каталога\n");

return 1;

}

//начиная с нужного каталога

while(dir1 = readdir(track))//открываем послед-но файлы и каталоги

{

stat(dir1->d\_name, &cur\_file);// в файл cur\_file информация о текущем файле(мб каталоге)

if(S\_ISDIR(cur\_file.st\_mode))//если это каталог

{

//printf("Да, это каталог и его зовут %s\n", dir1->d\_name);

subdir = opendir(dir1->d\_name);//пытаемся прочитать подкаталог

if(!subdir) printf("нет прав для чтения %s\n",dir1->d\_name);

else

{

readdir(subdir);

flag = chdir(dir1->d\_name); //dir1 - текущий каталог

if(flag) printf("ошибка при откртии каталога %s\n", dir1->d\_name);

else//проверка на подкаталоги

{

flag = 0;//предположим, что нет подкаталогов

readdir(subdir);

readdir(subdir);

while(dir2 = readdir(subdir))

{

stat(dir2->d\_name,&cur\_file);

if(S\_ISDIR(cur\_file.st\_mode))

{

flag = 1;//каталог содержит подкаталог

//printf("От %s\tесть каталог %s\n", dir1->d\_name ,dir2->d\_name);

break;

}

}

if(flag == 0)//каталог не содержал

printf("каталог, который не содежит подкаталогов:%s\n", dir1->d\_name);

closedir(subdir);

chdir("..");

}

}

}

}

closedir(track);

return 0;

}