#include <stdio.h>

#include <sys/types.h>

#include <dirent.h>

#include <string.h>

#include <pthread.h>

void \*Library(void \* argv);

void \*Checar(void \* argv);

int global = 0;

int return\_value;

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

{

int contador = 0;

DIR \*dir;

struct dirent \*drnt;

// excepcion de si el usuairo no ingresa un directorio

if (argc < 2){

printf("Usage: %s <directory>\n", argv[0]);

return -1;

}

// se abre el directorio en el indice 1, se empieza en 0 pero este no se toma en cuenta pues es el mismo

dir = opendir(argv[1]);

if (dir == NULL){ // en caso de no existir, se le indica al usuario que el argumento ingresado no es valido

printf("%s: Not a valid directory\n", argv[1]);

return -1;

}

//crea un thread por cada directorio que encuentra

while ((drnt = readdir(dir)) != NULL){

pthread\_t tid; //identificador

pthread\_attr\_t attr; // atributos del thread

pthread\_attr\_init(&attr); //atr por default

//creamos un thread recursivo

return\_value = pthread\_create(&tid,&attr,Library, (void\*)drnt);

if(return\_value < 1){ //revisamos si hubo error

printf("Fallo al crear thread");

continue;

}

pthread\_join(tid, NULL);

printf("Se finalizo un thread");

}

closedir(dir);

printf("Number of files is: %d\n",global);

}

//se invoca esta funcion cuando se tiene otro directorio, la cual crea otro thread para contar

void \*Checar(void \* argv){

DIR \*dir;

struct dirent \*drnt;

dir = opendir(argv);

pthread\_t tid; //identificador

pthread\_attr\_t attr; // atributos del thread

pthread\_attr\_init(&attr); //atr por default

//creamos un thread recursivo

return\_value = pthread\_create(&tid,&attr,Library,(void\*) drnt); //manda el direcotrio y Library se encarga de conntar

if(return\_value < 1){

printf("Fallo al crear thread");

}

pthread\_join(tid, NULL);

}

//funcion que cuenta los archivos

void \*Library(void \* argv){

DIR \*dir;

struct dirent \*drnt;

while ((drnt = readdir(dir)) != NULL){ //leemos los archivos

if(drnt->d\_type == DT\_DIR){

Checar((void\*)drnt); //en caso de ser un directorio, llamamos a la funcion checar para crear a otro thread

}

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

continue; //Skip current directory and parent directory/

}

if(drnt->d\_type == DT\_REG){ // en caso de ser directorio, le suma 1 al contador

global ++;

}

closedir(drnt); //cerramos el directorio

}

}