#include "lsdir.h"

void getPermissions(struct stat \*st, char \* p){

int perm = st->st\_mode;

//type

(S\_ISDIR(perm)) ? (p[0]='d') : (p[0]='-');

//USER

(perm & S\_IRUSR) ? (p[1]='r') : (p[1]='-');

(perm & S\_IWUSR) ? (p[2]='w') : (p[2]='-');

(perm & S\_IXUSR) ? (p[3]='x') : (p[3]='-');

//GROUP

(perm & S\_IRGRP) ? (p[4]='r') : (p[4]='-');

(perm & S\_IWGRP) ? (p[5]='w') : (p[5]='-');

(perm & S\_IXGRP) ? (p[6]='x') : (p[6]='-');

//OTHER

(perm & S\_IROTH) ? (p[7]='r') : (p[7]='-');

(perm & S\_IWOTH) ? (p[8]='w') : (p[8]='-');

(perm & S\_IXOTH) ? (p[9]='x') : (p[9]='-');

}

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

char filepath[PATH\_MAX+1];

DIR \*diretorio;

struct dirent \*d;

struct stat fs;

int file,tmp\_files;

if (argc != 2){

printf("WRONG NUMBER OF ARGUMENTS\n");

exit(1);

}

if ((diretorio = opendir (argv[1])) == NULL)

exit(1);

if((file= open("files.txt",O\_WRONLY | O\_CREAT | O\_SYNC | O\_TRUNC | O\_APPEND,0644)) == -1) //sorted files

exit(1);

if((tmp\_files= open("tmp\_files.txt",O\_WRONLY | O\_CREAT | O\_SYNC |O\_APPEND,0644)) == -1) //not sorted

exit(1);

//reach for every file in the path folder

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

{

if(sprintf(filepath,"%s/%s",argv[1],d->d\_name) == -1)

exit(1);

if(lstat(filepath,&fs) == -1)

exit(1);

//if the file is regular

if(S\_ISREG(fs.st\_mode) && !S\_ISDIR(fs.st\_mode)){

char \*name = d->d\_name;

char time[18];

char p[11];

char tmp[strlen(name)+37 + strlen(filepath)];

getPermissions(&fs,p);

if(strftime(time,18,"%y\_%m\_%d\_%T",localtime(&fs.st\_mtime)) == -1)

exit(1);

if(sprintf(tmp,"%s#%s#%s#%d#%s\n",name,time, p,(int)fs.st\_ino, filepath) == -1) //if they are separated by "#" we can detect files names with spaces

exit(1);

lseek(file,0,SEEK\_END);

if(write(tmp\_files,tmp,strlen(tmp)) == -1)

exit(1);

}

}

dup2(file,STDOUT\_FILENO);

closedir (diretorio);

close(file);

close(tmp\_files);

execlp("sort","sort","-r","tmp\_files.txt",NULL);

return 0;

}