

tree.c, Desplegando el tipo de archivo.

```
Terminal - alejandro@alejandro-VirtualBox: ~/Escritorio/Lab5
Archivo Editar Ver Terminal Pestañas Ayuda
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ls
hola prueba tree tree.c
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ gcc -o tree tree.c
tree.c: In function 'tree':
tree.c:112:25: warning: implicit declaration of function 'strcmp' [-Wimplicit-function-declaration]
    112 |         if (!strcmp(".", dp->d_name) || !strcmp("../", dp->d_name))
        |                         ^
tree.c:100:34: warning: 'sprintf' may write a terminating nul past the end of the destination [-Wformat-overflow=]
    100 |         sprintf(archivo, "%s/%s", path, dp->d_name);
        |                                ^
tree.c:100:12: note: 'sprintf' output 2 or more bytes (assuming 257) into a destination of size 256
    100 |         sprintf(archivo, "%s/%s", path, dp->d_name);
        |         ^~~~~~
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ls
hola prueba tree tree.c
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ./tree hola
.
..
hola1
    hola11
        ..
        .
        hola12
            ..
            .
        .
    h1
hola2
    ..
    .
    h2
.
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$
```

```
Terminal - alejandro@alejandro-VirtualBox: ~/Escritorio/Lab5
Archivo Editar Ver Terminal Pestañas Ayuda
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ./tree -f hola
(o) h
.. (d)
hola1 (d)
    hola11 (d)
        .. (d)
        . (d)
        .. (d)
    hola12 (d)
        .. (d)
        . (d)
    . (d)
    (o) h1
hola2 (d)
    .. (d)
    . (d)
    (o) h2
. (d)
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$
```

El código que proporcionaste es un programa en C que implementa una versión simplificada del comando "tree" en sistemas Unix. El programa toma un directorio como argumento en la línea de comandos y muestra una representación en forma de árbol de la estructura de directorios y

archivos en ese directorio. Además, tiene una opción ("-f") que, cuando se activa, muestra información adicional sobre los archivos, como su tipo.

En tu salida, puedes ver la estructura de directorios y archivos en el directorio especificado, con la indicación que muestra la jerarquía de directorios. También puedes ver si una entrada es un directorio o un archivo y, en el caso de los archivos, su tipo.

Estado1.c

```
Terminal - alejandro@alejandro-VirtualBox: ~/Escritorio/Lab5
Archivo Editar Ver Terminal Pestañas Ayuda
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ gcc -o estado1 Estado1.c
Estado1.c: In function 'estado':
Estado1.c:33:38: warning: format '%d' expects argument of type 'int', but argument 3 has type '__ino_t' {aka 'long unsigned int'} [-Wformat=]
33 |         fprintf(stdout, " Nro. de inode: %d\n", buf.st_ino);
    |                                ^~
    |                                |
    |                                int      __ino_t {aka long unsigned int}
    |                                %ld
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ls
estado1 Estado1.c hola prueba tree tree.c
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ./estado1 tree.c
argc=2
Archivo: tree.c
Nro. de inode: 1056529
Tipo:ordinario
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ./estado1 hola
argc=2
Archivo: hola
Nro. de inode: 1056533
Tipo:directorio
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$ ls -l
1056542 estado1 1056546 Estado1.c 1056533 hola 1056538 prueba 1056532 tree 1056529 tree.c
alejandro@alejandro-VirtualBox:~/Escritorio/Lab5$
```

"estado1.c" se utiliza para mostrar información estadística sobre uno o varios archivos.

Finalmente, se utiliza el comando "ls -l" para mostrar los números de inodo de los archivos en el directorio actual, y puedes ver que los números de inodo coinciden con los que se muestran en la salida del programa para "tree.c" y "hola".

Copiar.c

```
Terminal - alejandro@alejandra-VirtualBox: ~/Escritorio/Lab5
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ls
Copiar.c  estado1  Estadol.c  hola  original.txt  prueba  tree  tree.c
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ gcc -o copiar Copiar.c
Copiar.c: In function 'main':
Copiar.c:44:33: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
   44 |         while ((nbytes=read(fd_origen, buffer, sizeof buffer))> 0)
      |                               ^~~~~~
      |                               fread
Copiar.c:45:34: warning: implicit declaration of function 'write'; did you mean 'fwrite'? [-Wimplicit-function-declaration]
   45 |         write(fd_destino, buffer, nbytes);
      |         ^~~~~~
      |         fwrite
Copiar.c:47:18: warning: implicit declaration of function 'close'; did you mean 'pclose'? [-Wimplicit-function-declaration]
   47 |         close(fd_origen);
      |         ^~~~~~
      |         pclose
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ./copiar original.txt copia.txt
Num. descriptor del origen 3
Num. descriptor del destino 4
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$
```

copia el contenido de un archivo de origen en un archivo de destino.

```
Terminal - alejandro@alejandra-VirtualBox: ~/Escritorio/Lab5
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ls
copiar  Copiar.c  copia.txt  estado1  Estadol.c  hola  original.txt  prueba  tmp  tree  tree.c
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ln tmp tmp1
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ls -ll
total 84
-rwxrwxr-x 1 alejandro alejandro 17072 oct 27 15:01 copiar
-rw-rw-r-- 1 alejandro alejandro 1826 oct 27 14:59 Copiar.c
-rw-rw-r-- 1 alejandro alejandro 0 oct 27 15:01 copia.txt
-rwxrwxr-x 1 alejandro alejandro 17200 oct 27 14:54 estado1
-rw-rw-r-- 1 alejandro alejandro 2092 oct 27 14:53 Estadol.c
-rw-rw-r-- 1 alejandro alejandro 4096 oct 27 14:45 hola
drwxrwxr-x 4 alejandro alejandro 4096 oct 27 14:45 hola
-rw-rw-r-- 1 alejandro alejandro 0 oct 27 15:00 original.txt
drwxrwxr-x 2 alejandro alejandro 4096 oct 27 14:41 prueba
-rw-rw-r-- 2 alejandro alejandro 0 oct 27 15:07 tmp
-rw-rw-r-- 2 alejandro alejandro 0 oct 27 15:07 tmp1
-rwxrwxr-x 1 alejandro alejandro 17424 oct 27 14:48 tree
-rw-rw-r-- 1 alejandro alejandro 5404 oct 27 14:31 tree.c
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ rm tmp
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$ ls -ll
total 84
-rwxrwxr-x 1 alejandro alejandro 17072 oct 27 15:01 copiar
-rw-rw-r-- 1 alejandro alejandro 1826 oct 27 14:59 Copiar.c
-rw-rw-r-- 1 alejandro alejandro 0 oct 27 15:01 copia.txt
-rwxrwxr-x 1 alejandro alejandro 17200 oct 27 14:54 estado1
-rw-rw-r-- 1 alejandro alejandro 2092 oct 27 14:53 Estadol.c
drwxrwxr-x 4 alejandro alejandro 4096 oct 27 14:45 hola
-rw-rw-r-- 1 alejandro alejandro 0 oct 27 15:00 original.txt
drwxrwxr-x 2 alejandro alejandro 4096 oct 27 14:41 prueba
-rw-rw-r-- 1 alejandro alejandro 0 oct 27 15:07 tmp1
-rwxrwxr-x 1 alejandro alejandro 17424 oct 27 14:48 tree
-rw-rw-r-- 1 alejandro alejandro 5404 oct 27 14:31 tree.c
alejandra@alejandra-VirtualBox:~/Escritorio/Lab5$
```

Se puede ver que se creó un enlace simbólico llamado "tmpl" al archivo "tmp". Luego, se eliminó el archivo "tmp", lo que se refleja en la lista de archivos en el directorio.

eliminar el archivo original (tmp) no afecta al enlace duro (tmpl). Los datos siguen siendo accesibles a través del enlace duro, y el inodo no se libera hasta que se eliminen todos los enlaces duros a él.