

# Gestor de archivos

---

LENGUAJE ENSAMBLADOR

Fernando Torres Bautista

UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA | INGENIERIA EN COMPUTACIÓN

## 1. Proyecto final

Se propone realizar un gestor que realice lo siguiente:

- Poder navegar entre directorios
- Mostrar el contenido de las carpetas
- Crear/Borrar/Renombrar archivos
- Crear/Borrar/Renombrar directorios

La propuesta solo señala las acciones que debe realizar el programa.

## 2. Manual de usuario

El usuario podrá ejecutar el programa dando doble clic sobre el ejecutable, se hizo y probó en el TASM, si se desea ensamblar y ligar se necesitan el archivo Proy.asm y fun.asm.

Primero: tasm Proy

Segundo: tlink Proy fun

Al ser ejecutado, se mostrará un menú con nueve opciones que va del 0 al 8, en función a la tarea que se desea realizar. Si el usuario pulsa cualquier tecla fuera del menú se volverá a mostrar el menú principal.

Si el usuario elije la opción 0: Se ejecutara la opción de desplegar el contenido del directorio actual mostrando archivos y subdirectorios por igual.

Si el usuario elije la opción 1: En la opción de "CD" se tiene que el directorio donde se ejecute el programa es el principal o raíz, para ingresar a un subdirectorio basta con ingresar el nombre de éste. Para regresar al subdirectorio anterior se tiene que poner la ruta desde la raíz, ejemplo: si mi unidad está señalada como "d:/" y estoy en un subdirectorio "d:>carpeta1>final", y quiero regresar a "carpeta1", se tendría que dar lo siguiente d:/carpeta1 como subdirectorio a la acción. Si se desea retornar directamente al directorio raíz, basta con escribir "/".

Si el usuario elije la opción 2: La opción te pedirá un nombre para el subdirectorio a crear. Posteriormente te mostrará el menú principal. Para salir de esta acción solo es necesario pulsar enter y se mostrará un mensaje de error al no poder crear el subdirectorio con una cadena vacía como nombre.

Si el usuario elije la opción 3: La opción te pedirá el nombre del subdirectorio a eliminar. Esta acción se realiza si el subdirectorio está vacío, de lo contrario mostrará un mensaje de error.

Para cancelar la acción basta con dar un enter, al ser vacía la cadena introducida no realizará ninguna acción y solo se mostrará un mensaje de error al no crear el subdirectorio.

Si el usuario elije la opción 4: Esta opción renombra un subdirectorio, primero te pide el nombre actual del subdirectorio, y posteriormente el nuevo nombre para el subdirectorio. Si no se desea realizar esta acción se deberá dar un nombre inexistente de carpeta a renombrar, para que no se de la acción y solo retorne un error al no encontrar el subdirectorio especificado para realizar el renombramiento.

Si el usuario elije la opción 5: La opción te pedirá un nombre para el archivo a crear, cabe destacar que se debe incluir la extensión del mismo. Posteriormente te mostrará el menú principal. Para salir de esta acción solo es necesario pulsar enter y se mostrará un mensaje de error al no poder crear el archivo con una cadena vacía como nombre.

Si el usuario elije la opción 6: La opción te pedirá el nombre del archivo a eliminar, cabe destacar que se debe incluir la extensión del archivo a eliminar. Para cancelar la acción basta con dar un enter, al ser vacía la cadena introducida no realizará ninguna acción y solo se mostrará un mensaje de error al no poder eliminar un archivo con una cadena vacía.

Si el usuario elije la opción 7: Esta opción renombra un archivo, primero te pide el nombre actual y posteriormente el nuevo nombre para el archivo. Si no se desea realizar esta acción se deberá dar un nombre inexistente de archivo a renombrar, para que no se de la acción y solo retorne un error al no encontrar el archivo especificado para realizar el renombramiento.

Si el usuario elije la opción 8, simplemente se saldrá de la ejecución del programa.

### 3. Manual técnico

El programa está hecho con dos archivos, proy1.asm el archivo principal del programa y el archivo fun.asm que contiene algunas funciones que son necesarias para el programa.

#### 3.1 proy1.asm

Hace uso de dos segmentos, el de datos, y el de código. En el segmento de datos hemos definido lo necesario para usar a lo largo del programa, mientras que en el de código hemos realizado los algoritmos necesarios para el mismo.

El programa comienza con la definición de una única macro para imprimir en pantalla. Delante de ésta macro se define el main del programa que después de obtener la dirección del segmento de datos y ponerla en el registro de bloque de datos y el extra, se define buclePri, El bucle principal del programa donde nos muestra el menú de control, para darnos opciones como el desplegar el directorio actual (etiqueta dir\_ac), navegar por los subdirectorios con cd (etiqueta cd\_cha), crear subdirectorios (etiqueta crea\_dir), crear archivos (etiqueta crea\_fic), eliminar subdirectorios (etiqueta borrar\_dir), borrar archivo (etiqueta borrar\_fic), renombrar subdirectorio (etiqueta renombra\_dir), renombrar archivos (etiqueta renombra\_fic) y por último la etiqueta de salida (etiqueta out1). Para acceder a dichas opciones una vez que se muestre el menú se procederá a leer un carácter del teclado, posteriormente comparando con el registro "AL" con los valores de las opciones, se salta a la etiqueta correspondiente. Si la opción introducida por el teclado no es ninguna de las descritas por el menú, no se realiza ninguna acción y se muestra nuevamente el menú principal.

En función a la opción que el usuario elija se saltará a una etiqueta u a otra, cada una de las etiquetas regresa al menú principal a excepción de salir:

- Etiqueta dir\_ac:  
Esta opción llama a la función dir de fun, la cual hace uso de la función 47h interrupción 21h para obtener la dirección actual, la despliega. Después establece la memoria del DTA mediante la interrupción 21h función 1Ah, establecemos el patrón de búsqueda y así se busca el primer archivo que cumpla con el patrón. El tercer paso es el ciclo para desplegar mientras existan más archivos con el mismo patrón. Se sale del ciclo si detecta un bit de acarreo.
- Etiqueta cd\_cha:  
Esta opción primero despliega en pantalla un mensaje haciendo uso de la macro print, posteriormente lee del teclado la ruta a la cual acceder. Si se desea entrar en un subdirectorio basta con dar el nombre. Si se desea regresar al anterior se tiene que escribir la ruta completa desde la unidad en la que se ejecute el programa. Si se desea regresar al directorio raíz basta con pulsar "/". Ésta acción se realiza por medio de la interrupción 21h función 3Bh, cargando en el registro DX la ruta. Si se ejecuta de forma adecuada regresa al menú, si no antes te muestra un mensaje de error.
- Etiqueta crea\_dir:

Esta opción despliega en pantalla un mensaje haciendo uso de la macro print, pidiendo el nombre del subdirectorio a crear. Ésta acción se realiza por medio de la interrupción 21h función 39h, cargando en el registro DX el nombre del subdirectorio. Si se ejecuta de forma adecuada regresa al menú, si no por medio de la bandera de acarreo antes muestra un mensaje de error pasando por la etiqueta error\_1.

- Etiqueta borrar\_dir:

Esta opción despliega en pantalla un mensaje haciendo uso de la macro print, pidiendo el nombre del subdirectorio a eliminar. Haciendo uso de la interrupción 21h función 3Ah, cargando en el registro DX el nombre del subdirectorio a eliminar. Si se ejecuta de forma adecuada regresa al menú, si no por medio de la bandera de acarreo antes muestra un mensaje de error en la etiqueta error\_2.

- Etiqueta renombra\_dir:

Esta opción despliega en pantalla 2 mensajes, en el primero te pide el nombre del subdirectorio a renombrar, después por medio de las funciones cadNombreA y después la función delimitarNA obtiene el nombre. Una vez obtenida la cadena del teclado compara su tamaño, si es igual a 0 se retorna a la etiqueta renombra\_dir para poder comenzar con el proceso nuevamente. En el segundo mensaje te pide el nuevo nombre, usando las funciones cadNombre y delimitarN, que lo mismo que las dos antes descritas pero para diferentes datos declarados. Una vez leída la segunda cadena, se compara su tamaño con 0 y nuevamente retorna a la etiqueta renombra\_dir si ésta es igual a 0. De pasar ambos mensajes, se realiza la acción de renombramiento por medio de la interrupción 21h función 56h, colocando en el registro DX el nombre del directorio a renombrar y en DI el nuevo nombre. Si se ejecuta correctamente se regresa al bucle principal, si no se realiza la acción de forma correcta muestra un mensaje de error y después retorna al bucle principal.

- Función leecad:

A continuación se muestra en el código la función leecad, hace la acción de recoger una cadena del teclado y acomodarla en un bufer, declarada en el segmento de datos, también delimita la cadena.

- Función cadNombre:

Haciendo uso de la interrupción 21h función 0Ah, recoge la cadena introducida desde el teclado, y después la función delimitarN.

- Función cadNombreA:

Haciendo uso de la interrupción 21h función 0Ah, recoge la cadena introducida desde el teclado, y después la función delimitarNA.

- Función delimitarN:

Termina la cadena con 00h para poder renombrar y pone el delimitador "\$" de exhibición de caracteres.

- **Función delimitarNA:**  
Termina la cadena con 00h para poder renombrar y pone el delimitador “\$” de exhibición de caracteres.
- **Etiqueta crea\_fic:**  
Esta opción despliega en pantalla un mensaje haciendo uso de la macro print, pidiendo el nombre del archivo a crear agregando la extensión de éste, una vez obtenido el nombre compara que su longitud sea mayor a cero para que no sea una cadena vacía. Ésta acción se realiza por medio de la interrupción 21h función 3Ch, cargando en el registro DX el nombre del archivo, si no se crea de forma correcta por medio de la bandera de acarreo se muestra el mensaje de error\_4. Si se ejecuta de forma adecuada muestra un mensaje de que el archivo se ha creado, cierra el archivo en este punto y se comprueban errores con la bandera de acarreo (jc) nuevamente, después regresa al bucle principal.
- **Etiqueta borrar\_fic:**  
Esta opción despliega en pantalla un mensaje haciendo uso de la macro print, pidiendo el nombre del archivo a eliminar, de la misma forma que al crear comprueba que la cadena no sea vacía. Haciendo uso de la interrupción 21h función 41h, cargando en el registro DX el nombre del archivo a eliminar, recordando que se debe agregar la extensión del archivo. Si se ejecuta de forma adecuada regresa al menú, muestra un mensaje de éxito, si no por medio de la bandera de acarreo antes muestra un mensaje de error en la etiqueta error\_5.
- **Etiqueta renombra\_fic:**  
Esta opción despliega en pantalla 2 mensajes, en el primero te pide el nombre del archivo a renombrar, después por medio de las funciones cadNombreA y después la función delimitarNA obtiene la cadena. Una vez obtenida la cadena del teclado compara su tamaño, si es igual a 0 se retorna a la etiqueta renombra\_dir para poder comenzar con el proceso nuevamente. En el segundo mensaje te pide el nuevo nombre para el archivo, usando las funciones cadNombre y delimitarN, que lo mismo que las dos antes descritas pero para diferentes datos declarados. Una vez leída la segunda cadena, se compara su tamaño con 0 y nuevamente retorna a la etiqueta renombra\_dir si ésta es igual a 0. De pasar ambos mensajes, se realiza la acción de renombramiento por medio de la interrupción 21h función 56h, colocando en el registro DX el nombre del archivo a renombrar y en DI el nuevo nombre. Si se ejecuta correctamente se regresa al bucle principal, si no se realiza la acción de forma correcta muestra un mensaje de error de la etiqueta error\_6 y después retorna al bucle principal.

### 3.2 fun.asm

El archivo fun.asm consta de dos segmentos en uso el de datos y el de código. En el segmento de datos se tienen los parámetros necesarios para que la función dir funciones correctamente.

El archivo consta de 4 funciones:

- dir: Muestra en pantalla el contenido del directorio actual.
- despc: Despliega el contenido de una cadena, terminada con cero
- reto: Da un salto de línea
- lee1: lee un carácter del teclado y lo retorna al.

### 3.3 Código proy.asm

.model small

extrn reto:near

extrn lee1:near

extrn dir:near

.386

.stack

.data

;Para las funciones dir

buf label byte ; Lista de parametros del nombre:

max\_blen db 80 ; \* long max del nombre

bufer\_len db ? ; \* #n de caracteres introducidos

bufer db 81 DUP(' '); \* nuevo nombre del fichero

;Para el renombramiento

nuevopar label byte ; Lista de parametros del nombre:

max\_nlen db 80 ; \* long max del nombre

nuevo\_len db ? ; \* #n de caracteres introducidos

nuevo db 81 DUP(' '); \* nuevo nombre del fichero

viejopar label byte ; Lista de parametros del nombre:

max\_vlen db 80 ; \* long max del nombre

viejo\_len db ? ; \* # n de caracteres introducidos

viejo db 81 DUP(' '); \* nombre original del fichero

.code

;Macros usadas en el programa

print macro cadena

local dbcad,dbfin,salta

pusha ;respalda todo ax,bx,cx,dx

push ds ;respalda DS, porque vamos a usar segmento de código

mov dx,cs ;segmento de codigo será también de datos

mov ds,dx

mov dx,offset dbcad ;direccion de cadena (en segmento de código)

mov ah,09h

int 21h ;desplegar

jmp salta ;saltar datos para que no sean ejecutados

dbcad db cadena ;aqui estara la cadena pasada en la sustitucion

dbfin db 24h ;fin de cadena

salta: pop ds ;etiqueta local de salto, recuperar segmento de datos

```

        popa                ;recuperar registros, opuesto a pusha
endm    print

```

main:

```

    mov ax,@data
    mov ds,ax
    mov es,ax

```

;Bucle principal

buclePri:

```

    ;Menú del gestor
    print

```

```

"*****"

```

```

    call reto
    print "* Gestor de archivos                *"
    call reto
    print "* 0. Dir   : Despliega el contenido del directorio actual        *"
    call reto
    print "* 1. Cd    : Para regresar al directorio raiz ----> '/'            *"
    call reto
    print "* 2. Crear subdirectorio  5. Crear archivo                        *"
    call reto
    print "* 3. Eliminar subdirectorio      6. Eliminar archivo                *"
    call reto
    print "* 4. Renombrar subdirectorio    7. Renombrar archivo                *"
    call reto
    print "*                               8. Salir                            *"
    call reto
    print

```

```

"*****"

```

```

    call reto

```

;Lectura de la opción escogida

```

    call lee1        ;resultado en la parte baja del registro ax, [al]
    call reto

```

;Comparación para definir lo que debe realizar el programa

;Primero lo de los directorios

```

    cmp al,0        ; Dir, desplegar el contenido actual del directorio
    je dir_ac       ; directorio actual

```

```

    cmp al,1        ; Cd, para entrar en un subdirectorio
    je cd_cha       ; cd cambiar directorio

```

```

    cmp al,2        ; Para crear subdirectorio
    je crea_dir     ;

```



```

    cmp al,3      ; Para borrar un directorio
    je borrar_dir

    cmp al,4      ; Para renombrar un directorio
    je renombra_dir

;Ahora los archivos
    cmp al,5      ; Para crear un archivo
    je crea_fic

    cmp al,6      ; Para borrar un archivo
    je borrar_fic

    cmp al,7      ; Para renombrar un archivo
    je renombrar_fic

    cmp al,8      ; Para salir del programa
    je out1
    jmp buclePri
out1:
    .exit 0

;Para los directorios las funciones y saltos
;-----
dir_ac:
    call dir ;llamo a la función dir de fun
    call reto
    jmp buclePri ; regresamos a el inicio del gestor

cd_cha:
    print "Introduzca el nombre del subdirectorio: "
    call leecad
    mov dx,offset bufer
    mov ah,3bh
    int 21h
    jc error_cd
    jmp buclePri

error_cd:
    call reto
    print "Ocurrio un error al acceder al directorio."
    call reto
    jmp buclePri

;-----

crea_dir:

```

```

    print "Introduzca el nombre del directorio a crear: "
    call leecad          ;en dx se encuentra tamb
    mov dx,offset bufer
    mov ah, 39h
    int 21h
    jc error_1
    jmp buclePri

```

```

error_1:
    call reto
    print "Ocurrio un error al crear el directorio."
    call reto
    jmp buclePri

```

```

borrar_dir:
    print "Introduzca el nombre del directorio a borrar: "
    call leecad          ;el nombre en el bufer
    mov dx,offset bufer
    mov ah,3Ah
    int 21h
    jc error_2
    jmp buclePri

```

```

error_2:
    call reto
    print "Error al eliminar el subdirectorio."
    call reto
    jmp buclePri

```

```

renombrar_dir:
    print "Introduce el nombre del directorio a renombrar: "
    call cadNombreA
    call delimitarNA
    call reto

    cmp viejo_len,00
    je renombrar_dir

    print "Introduce el nuevo nombre del directorio: "
    call cadNombre          ;cadena del nombre nuevo
    call delimitarN
    call reto

    cmp nuevo_len,00        ;si no se escribe ningun nombre
    je renombrar_dir

```

```

mov ah,56h          ;Funci n que renombra o mueve
mov dx,offset viejo ;Directorio o fiche a renombrar

```

```

mov di,offset nuevo      ;Renombra el dir o fich con la nueva cadena
int 21h                  ;introducida por teclado.
jc error_3
jmp buclePri

```

```

error_3:
    call reto
    print "Se produjo un error al renombrar."
    call reto
    jmp buclePri

;

```

```

leecad:
    mov ah,0Ah
    mov dx,offset buf     ;recoge la cadena a renombrar, desde teclado
    int 21h

    mov bh,00
    mov bl,buffer_len     ;Delimita cadena introducida a tamaño namelen
    mov buffer[bx],00h    ;Termina cadena con 00h para poder renombrar
    mov buffer[bx+1],'$'   ;Pone delimitador de exhibición de caracteres
    ret

```

;Para renombrar un directorio

```

cadNombre:
    mov ah,0Ah
    mov dx,offset nuevopar ;recoge la cadena a renombrar, desde teclado
    int 21h
    ret

```

```

cadNombreA:
    mov ah,0Ah
    mov dx,offset viejopar ;recoge cadena original desde teclado
    int 21h
    ret

```

```

delimitarN:
    mov bh,00
    mov bl,nuevo_len      ;Delimita cadena introducida a tamaño namelen
    mov nuevo[bx],00h     ;Termina cadena con 00h para poder renombrar
    mov nuevo[bx+1],'$'   ;Pone delimitador de exhibición de caracteres
    ret

```

```

delimitarNA:
    mov bh,00
    mov bl,viejo_len      ;Delimita cadena introducida a tamaño viejo_len
    mov viejo[bx],00h     ;Termina cadena con 00h para poder renombrar

```

```
mov viejo[bx+1],'$';Pone delimitador de exhibici n de caracteres
ret
```

```
;-----
```

```
;Para archivos se tiene lo siguiente
```

```
;En la creacion de los archivos se define la extension como parte del nombre
```

```
crea_fic:
```

```
print "Introduce el nombre del archivo: "
```

```
call cadNombre ;cadena del nombre nuevo
```

```
call delimitarN
```

```
call reto
```

```
cmp nuevo_len,00 ;si no se escribe ningun nombre se retorna al ciclo
```

```
je error_4
```

```
;crear
```

```
mov ah,3ch
```

```
mov cx,0
```

```
mov dx,offset nuevo
```

```
int 21h
```

```
jc error_4 ;si no se pudo crear
```

```
call reto
```

```
print " Fichero creado con  xito."
```

```
call reto
```

```
mov bx,ax
```

```
mov ah,3eh ;cierra el archivo
```

```
int 21h
```

```
jc error_4
```

```
jmp buclePri
```

```
error_4:
```

```
call reto
```

```
print "Se produjo un error al crear el archivo."
```

```
call reto
```

```
jmp buclePri
```

```
;Para borrar un archivo se tiene que escribir la extension como parte del nombre
```

```
borrar_fic:
```

```
print "Introduce el nombre del archvio a borrar: "
```

```
call cadNombre ;cadena del nombre dell archivo a borrar
```

```
call delimitarN
```

```
call reto
```

```
cmp nuevo_len,00 ;si no se escribe ningun nombre se retorna al ciclo
```

```
je error_5
```

```
mov ah,41h
```

```
mov dx, offset nuevo
```

```
int 21h
```

```
jc error_5 ;Si hubo error
```

```
call reto
```

```
print "Archivo eliminado con exito"
call reto
jmp buclePri
```

```
error_5:
    call reto
    print "Se produjo un error al eliminar el archivo."
    call reto
    jmp buclePri
```

```
renombrar_fic:
    print "Introduce el nombre del archivo a renombrar: "
    call cadNombreA
    call delimitarNA
    call reto

    cmp viejo_len,00
    je renombrar_fic

    print "Introduce el nuevo nombre del archivo: "
    call cadNombre                ;cadena del nombre nuevo
    call delimitarN
    call reto

    cmp nuevo_len,00                ;si no se escribe ningun nombre
    je renombrar_fic

    mov ah,56h                    ;Funcion que renombra o mueve
    mov dx,offset viejo            ;Directorio o fiche a renombrar
    mov di,offset nuevo            ;Renombra el dir o fich con la nueva cadena
    int 21h                        ;introducida por teclado.
    jc error_6
    jmp buclePri
```

```
error_6:
    call reto
    print "Se produjo un error al renombrar."
    call reto
    jmp buclePri
```

;Para la creacion/renombramiento/eliminacion de archivos se debe incluir la extensiones de los mismos.

;El tamaño maximo es de 8 letras por nombre.

End

## 4. Ejecución

Al ejecutar el programa se muestra el menú de la siguiente forma:

```
D:\>proy1
*****
*      Gestor de archivos      *
* 0. Dir      : Despliega el contenido del directorio actual      *
* 1. Dd : Para regresar al directorio raiz ----> '/'              *
* 2. Crear subdirectorio      5. Crear archivo                    *
* 3. Eliminar subdirectorio    6. Eliminar archivo                *
* 4. Renombrar subdirectorio   7. Renombrar archivo                *
*                               8. Salir                            *
*****
```

A continuación se muestra un ejemplo total, primero hacemos un cd a la carpeta “car”:

```
*****
*      Gestor de archivos      *
* 0. Dir      : Despliega el contenido del directorio actual      *
* 1. Dd : Para regresar al directorio raiz ----> '/'              *
* 2. Crear subdirectorio      5. Crear archivo                    *
* 3. Eliminar subdirectorio    6. Eliminar archivo                *
* 4. Renombrar subdirectorio   7. Renombrar archivo                *
*                               8. Salir                            *
*****
1
Introduzca el nombre del subdirectorio: car_
```

Mostramos el contenido de ésta, con un Dir:

```
*****
0
CAR
.
..
*****
*      Gestor de archivos      *
* 0. Dir      : Despliega el contenido del directorio actual      *
* 1. Dd : Para regresar al directorio raiz ----> '/'              *
* 2. Crear subdirectorio      5. Crear archivo                    *
* 3. Eliminar subdirectorio    6. Eliminar archivo                *
* 4. Renombrar subdirectorio   7. Renombrar archivo                *
*                               8. Salir                            *
*****
```

Obsérvese que no hay contenido en la carpeta, más que el directorio . y .. ;

Para demostrar el funcionamiento procederemos creando, renombrando y eliminando un directorio y un archivo.

Primero creamos una carpeta con el nombre “micar”, después creamos un archivo con el nombre y extensión miarc.txt, posteriormente se despliega el contenido del directorio para verificar los cambios

```
*****
0
CAR
.
..
*****
*      Gestor de archivos      *
* 0. Dir      : Despliega el contenido del directorio actual      *
* 1. Dd : Para regresar al directorio raiz ----> '/'              *
* 2. Crear subdirectorio      5. Crear archivo                    *
* 3. Eliminar subdirectorio    6. Eliminar archivo                *
* 4. Renombrar subdirectorio   7. Renombrar archivo                *
*                               8. Salir                            *
*****
2
Introduzca el nombre del directorio a crear: micar_
```

```
*****
5
Introduce el nombre del archivo: miarc.txt
```

```
*****
0
CAR
.
..
MICAR
MIARC.TXT
*****
```

Ya que se han creado el archivo y directorio se renombran, y se muestran los cambios:

```
*****
4
Introduce el nombre del directorio a renombrar: micar
Introduce el nuevo nombre del directorio: carpe
*****
```

```
*****
7
Introduce el nombre del archivo a renombrar: miarc.txt
Introduce el nuevo nombre del archivo: archi.txt
*****
```

```
*****
0
CAR
.
..
CARPE
ARCHI.TXT
*****
```

Como se muestra en la imagen de la izquierda, la carpeta Car, contiene ahora el subdirectorio renombrado CARPE y el archivo archi.txt.

A continuación se muestran las ejecuciones de la opción eliminar para el archivo y el subdirectorio y se retorna al directorio raíz con "/" en la opción cd

```
*****
3
Introduzca el nombre del directorio a borrar: carpe
*****
```

```
*****
6
Introduce el nombre del archivo a borrar: archi.txt_
*****
```

```
*****
0
CAR
.
..
*****
```

Como se muestra en la imagen de la izquierda, el subdirectorio ha quedado completamente vacío.

Para regresar al subdirectorio raíz, opción 0, introducir "/"

```
0
CAR
.
..
*****
Gestor de archivos
*****
0. Dir      : Despliega el contenido del directorio actual
1. Dd      : Para regresar al directorio raíz --> '/'
2. Crear subdirectorio      5. Crear archivo
3. Eliminar subdirectorio    6. Eliminar archivo
4. Renombrar subdirectorio   7. Renombrar archivo
                               8. Salir
*****
Introduzca el nombre del subdirectorio: /
```

## 5. Bibliografía

[http://ict.udlap.mx/people/oleg/docencia/ASSEMBLER/asm\\_interrup\\_21.html](http://ict.udlap.mx/people/oleg/docencia/ASSEMBLER/asm_interrup_21.html)

Los procesadores Intel, Tercera edición, Barry B, Brey, Prentice Hall Hispanoamericana S.A., 1995