

# Tema IV Ejercicio III: Comandos DOS

Nicolás A. Ortega Froysa

28 de noviembre de 2021

# Índice

1. Hoja De Control Del Documento	3
2. Creación de Archivos y Directorios	4
3. Mover, Copiar, y Renombrar	5
4. Borrando Archivos y Directorios	5
5. Modificando los Atributos de un Archivo	6
6. Conclusión	6
7. Derechos de Autor y Licencia	7

# 1. Hoja De Control Del Documento

Cuadro 1: Documento/Archivo

<b>Fecha Última Modificación</b>	28/11/2021	<b>Versión/Revisión</b>	v01r02
<b>Fecha Creación</b>	28/11/2021		
<b>Fecha Finalización</b>	28/11/2021		

Cuadro 2: Registro De Cambios

<b>Versión/Revisión</b>	<b>Página(s)</b>	<b>Descripción</b>
v01r01	Todas	Creación y elaboración del documento.
v01r02	4-7	Añadir contenidos nuevos.

Cuadro 3: Autores Del Documento

<b>Apellidos, Nombre</b>	<b>Curso</b>
Ortega Froysa, Nicolás Andrés	1

<b>Preparado</b>	<b>Revisado</b>	<b>Aprobado</b>
Ortega Froysa, Nicolás Andrés		

```
Simbolo del sistema
C:\Users\TECH>cd practica
C:\Users\TECH\practica>tree .
Listado de rutas de carpetas
El número de serie del volumen es 0000000E 6875:8CFE
C:\Users\TECH\PRACTICA
├── carp1
│   ├── carp11
│   └── carp12
├── carp2
│   └── carp21
└── carp3
C:\Users\TECH\practica>
```

Figura 1: Comando `tree` de los archivos.

## 2. Creación de Archivos y Directorios

Para poder crear carpetas en el *shell* de DOS, usamos el comando `mkdir`. Esto también lo podemos usar para crear todo un *path* de directorios, uno dentro de otro, simplemente especificando el camino. De este modo podemos crear una estructura de directorio con los siguientes comandos:

```
>mkdir carp1\carp11
>mkdir carp1\carp12
>mkdir carp2\carp21
>mkdir carp3
```

Si ahora corremos el comando `tree` dentro de este directorio (que en nuestro caso se denomina `practica`) veremos toda la estructura de este directorio y sus subdirectorios (figura 1).

A partir de aquí podemos añadir archivos dentro de estos directorios. Añadiremos los archivos siguientes con cualquier editor de texto:

- `arch11.txt` en `carp11`
- `arch12.txt` en `carp12`
- `arch21.txt` en `carp21`

También se puede crear simplemente un archivo y copiarlo. Esto se podría hacer con el procedimiento siguiente, creando el primer archivo con el contenido “test” y copiándolo a los demás directorios con otro nombre:

```
# Crear archivo arch11.txt
>echo test >> carp1\carp11\arch11.txt
# Copiar el archivo primero a las distintas carpetas con otro
# nombre
>copy carp1\carp11\arch11.txt carp1\carp12\arch12.txt
>copy carp1\carp11\arch11.txt carp1\carp21\arch21.txt
```

### 3. Mover, Copiar, y Renombrar

Aquí acabamos de ver el comando `copy`, que sirve para copiar archivos. También existe otro que es `move` para mover los archivos. Estos funcionan con dos argumentos: el archivo original, y el archivo destino. En el caso del archivo destino, si se especifica un directorio ya existente en vez de un nombre de archivo para el nuevo destino, simplemente se copiará/moverá a ese directorio con el mismo nombre de archivo.

Como podemos imaginar con lo dicho anteriormente, el comando `move` se puede usar para cambiar el nombre de un archivo simplemente especificando el nuevo nombre como destino.

Cuando copiamos archivos de un lugar a otro, podemos concatenar el contenido de dos o más archivos a un nuevo archivo por medio del símbolo `+`. De este modo, el próximo comando concatenaría el contenido de `test0.txt` y `test1.txt` en un archivo `test01.txt`:

```
>copy test0.txt + test1.txt test01.txt
```

### 4. Borrando Archivos y Directorios

Para borrar archivos usamos el comando `del <archivo>`. Lo cual, para borrar el archivo `arch21.txt` podemos correríamos el comando siguiente:

```
>del carp2\carp21\arch21.txt
```

Si ya estamos dentro de esa carpeta, podemos simplemente correr `del arch21.txt`, o si queremos hacer referencia a él desde la raíz del dispositivo de almacenamiento (asumiendo que sea `C:`) correríamos:

```
>del C:\...\carp2\carp21\arch21.txt
```

Este comando no es posible usarlo directamente con directorios, ya que está hecho para trabajar sólo con archivos. Para borrar directorios podemos

usar el otro comando, `rmdir` que por defecto sólo funcionará si la carpeta está vacía. Para borrar una carpeta que tiene contenido dentro tendremos que usar el argumento `/s` de la forma siguiente.

```
>rmdir /s carp3
```

## 5. Modificando los Atributos de un Archivo

Para poder ver los atributos de los archivos y directorios que hemos creado podemos usar el comando `attrib [file]`. Este comando, si le añadimos el argumento `/s` se hará de forma recursiva (como vimos con `rmdir`) y nos mostrará todos los atributos de todos los archivos en un directorio. Si no especificamos ningún archivo por defecto mostrará todos los atributos de los archivos dentro del directorio actual. También podemos usar este mismo comando para modificar los atributos de un archivo por medio de un `+x` ó `-x` donde `x` es uno de los siguientes:

- `r`: sólo lectura (*read-only*)
- `h`: escondido (*hidden*)
- `s`: sistema (*system*)
- `a`: archivar (*archive*)

## 6. Conclusión

Usando la línea de comandos, también en Windows, nos proveerá más flexibilidad y poder sin necesidad de navegar un interminable entorno gráfico mal-organizado para encontrar las opciones que buscamos. En vez de esto, podemos simplemente usar el comando `help <comando>` para saber qué opciones tenemos para un comando, y de este modo también podremos finalizar nuestras tareas de una forma más rápida. Por ejemplo, la creación de los archivos al principio hubiera sido una tarea de varios minutos si usásemos un programa como *NotePad*, pero en la línea de comandos simplemente correremos un comando por cada archivo.

En fin, incluso en Windows usar la línea de comandos es más eficiente que usar el entorno gráfico, lo cual nos dice que seguramente deberían de haberse quedado mejor con MS-DOS.

## **7. Derechos de Autor y Licencia**

Copyright © 2021 Nicolás A. Ortega Froya <[nicolas@ortegas.org](mailto:nicolas@ortegas.org)>  
Este documento se distribuye bajo los términos y condiciones de la licencia  
Creative Commons Attribution No Derivatives 4.0 International.