

## *Sistemas Operativos*

### ***Comandos de procesamiento por lotes***

Un archivo o programa de procesamiento por lotes es un archivo de texto sin formato que contiene uno o más comandos de MS-DOS y que tiene asignado una extensión .BAT. Cuando escribe el nombre del programa de procesamiento por lotes en la línea de comandos, los comandos se ejecutan como un grupo.

Cualquier comando de MS-DOS que use en el símbolo del sistema también podrá incluirse en un programa de procesamiento por lotes. Los siguientes comandos de MS-DOS están diseñados especialmente para programas de procesamiento por lotes:

**CALL - CHOICE - ECHO - FOR - GOTO - IF - PAUSE - REM - SHIFT**

#### ***CALL***

Llama a un programa de procesamiento por lotes desde otro sin detener el funcionamiento del primer programa de procesamiento por lotes.

***CALL [unidad:][ruta] nombrearchivo [parámetros-lotes]***

#### ***Parámetros***

##### **[unidad:][ruta] nombrearchivo**

Especifica la posición y el nombre del programa de procesamiento por lotes que desee llamar. El nombre del archivo deberá contener una extensión .BAT.

##### **parámetros-lotes**

Especifica cualquier información para la línea de comandos que el programa de procesamiento por lotes requiera.

Parámetros-lotes podrá incluir cualquier información que se pueda suministrar a un programa de procesamiento por lotes, incluyendo modificadores, nombres de archivos, los parámetros reemplazables %1 a %9 y las variables de entorno tales como %baud%.

#### ***Uso de símbolos de canalización y redirección***

No utilice signos de canalización ("|") o de redirección ("<<", "<", ">", y ">>") con el comando CALL.

#### ***Llamadas recursivas***

Puede crear un programa de procesamiento por lotes que se llame a sí mismo, pero necesitará incluir una condición de salida. De lo contrario, podría crearse una operación de bucle infinita entre el programa de procesamiento por lotes padre y el programa de procesamiento por lotes hijo.

Para ejecutar el programa VERIFIC.BAT desde otro programa de procesamiento por lotes, deberá incluir el siguiente comando en el programa padre:

**call verifc**

Supongamos que el programa padre acepta dos parámetros reemplazables y Vd. desea que estos parámetros se pasen a VERIFIC.BAT. Podrá incluir el siguiente comando en el programa padre:

**call verifc %1 %2**

#### ***CHOICE***

En un programa de procesamiento por lotes, solicita del usuario que realice una selección. Realiza una solicitud y aguarda hasta que el usuario elija entre un conjunto de teclas específico. Sólo podrá utilizar este comando en programas de procesamiento por lotes.

**CHOICE [/C[:]teclas] [/N] [/S] [/T[:]c,nn] [texto]**

### **Parámetros**

#### **texto**

Especifica el texto que desea presentar antes del símbolo del sistema. Sólo se requieren comillas si incluye un carácter modificador (⌨) como parte del texto antes del símbolo del sistema. Si no especifica texto, CHOICE sólo presentará un símbolo del sistema.

### **Modificadores**

#### **/C[:]teclas**

Especifica las teclas permitidas en el símbolo del sistema. Al mostrarlas, dichas teclas estarán separadas por comas, aparecerán entre corchetes ([ ]) y estarán seguidas de un signo de interrogación. Si no especifica el modificador /C, CHOICE usará SN como la opción predeterminada. Los dos puntos (:) son opcionales.

#### **/N**

Evita que CHOICE presente el símbolo del sistema. Sin embargo, aún se presentará el texto especificado. Si especifica el modificador /N, las teclas especificadas aún serán válidas.

#### **/S**

Indica que CHOICE deberá distinguir entre letras mayúsculas y minúsculas. Si no se especifica el modificador /S, CHOICE aceptará tanto mayúsculas como minúsculas para las teclas que el usuario especifique.

#### **/T[:]c,nn**

Causa que CHOICE haga una pausa durante el número de segundos especificado antes de revertir a una tecla especificada. Los valores del modificador /T son los siguientes:

**c** Especifica a qué carácter revertirá después de haber transcurrido nn segundos. El carácter deberá encontrarse entre el conjunto de opciones especificado por el modificador /C.

**nn** Especifica el número de segundos que deberá pausar. Los valores aceptables están entre 0 y 99. Si se especifica 0, no se realizará la pausa antes de revertir.

### **Parámetros de ERRORLEVEL**

La primera tecla que asigne devolverá un valor de 1, la segunda un valor de 2, la tercera un valor de 3 y así sucesivamente. Si el usuario presiona una tecla que no se encuentre entre las teclas asignadas, CHOICE emitirá un sonido de advertencia (es decir, enviará un carácter BEL o 07h a la consola).

Si CHOICE detecta una condición de error, devolverá un valor ERRORLEVEL de 255. Si el usuario presiona CTRL+INTERRUMPIR o CTRL+C, CHOICE devolverá un valor ERRORLEVEL de 0.

Cuando utilice parámetros de ERRORLEVEL en un programa de procesamiento por lotes, lístelos en orden decreciente.

### **Lo que el usuario ve al usar CHOICE en un programa de procesamiento por lotes**

Si utiliza la siguiente sintaxis en un programa de procesamiento por lotes,

**choice /c:snc**

el usuario verá lo siguiente al iniciar CHOICE:

**[S,N,C]?**

Si le agrega texto a la sintaxis,

**choice /c:snc Sí, No o Continuar**

el usuario verá lo siguiente al iniciar CHOICE:

**Sí, No o Continuar [S,N,C]?**

### ***Lo que el usuario ve si se omite el símbolo del sistema***

Si, como se demuestra en el siguiente ejemplo, se utiliza el modificador /N para omitir el símbolo del sistema en un programa de procesamiento por lotes,

**choice /n ¿Sí, No o Continuar?**

el usuario verá sólo el texto especificado al iniciar CHOICE:

**¿Sí, No o Continuar?**

### ***Lo que el usuario ve si se usa el modificador /T***

Si utiliza la siguiente sintaxis en un programa de procesamiento por lotes,

**choice /c:snc /t:n,5**

el usuario verá lo siguiente al iniciar CHOICE:

**¿[S,N,C]?**

Si después de 5 segundos el usuario no ha presionado una tecla, CHOICE seleccionará N y devolverá un valor ERRORLEVEL de 2. Si el usuario presionó una tecla antes de los 5 segundos, CHOICE devolverá el valor que corresponda a la selección del usuario.

Para disponer de la opción de defragmentación de la unidad C cuando inicie su PC, agregue las siguientes líneas a su archivo AUTOEXEC.BAT:

```
choice Defrag unidad /ty,5  
if errorlevel 2 goto SkipDefrag  
defrag c:  
:SkipDefrag
```

Si presiona N antes de 5 segundos, DEFRAG no se ejecutará y CHOICE devolverá un valor ERRORLEVEL de 2. Si no presiona N antes de 5 segundos o si elige S, DEFRAG se ejecutará en la unidad C.

### ***Uso de CHOICE en un programa de procesamiento por lotes***

El siguiente programa de procesamiento por lotes demostrará el uso de la opción CHOICE para seleccionar uno de tres programas: Editor de MS-DOS, Microsoft Anti-Virus o Copia de seguridad (Backup).

Observe que las declaraciones IF ERRORLEVEL aparecen en orden decreciente. MS-DOS considerará que la declaración IF es veraz si el parámetro ERRORLEVEL devuelto por CHOICE es mayor o igual que el parámetro especificado en el comando IF.

```
@echo off  
cls  
echo.  
echo A Microsoft Editor  
echo B Microsoft Anti-Virus  
echo C Microsoft Copia de seguridad (Backup)  
echo.  
choice /c:abc Elija una opción  
if errorlevel 3 goto MSBackup  
if errorlevel 2 goto Msav  
if errorlevel 1 goto Edit  
  
:Edit  
edit  
goto End  
  
:Msav
```

**msav**  
**goto End**

**:Msbackup**  
**msbackup**  
**goto End**

**:End**

### ***ECHO***

Muestra u oculta el texto de programas de procesamiento por lotes cuando el programa se esté ejecutando. También indica si la función de repetición de comandos está activada o desactivada.

Al ejecutar un programa de procesamiento por lotes, MS-DOS generalmente presenta los comandos en la pantalla (hace eco) mientras se ejecutan. Es posible activar o desactivar esta característica mediante el comando ECHO.

<b><i>ECHO [ON/OFF]</i></b>
-----------------------------

Para usar el comando echo a fin de presentar un mensaje, utilice la sintaxis siguiente:

**echo [mensaje]**

### ***Parámetros***

#### **ON|OFF**

Especifica si será activada (on) o desactivada (off) la característica de presentar los comandos en la pantalla. Si desea ver el estado actual del comando ECHO, utilice dicho comando sin parámetro.

#### **mensaje**

Especifica el texto que desee que MS-DOS presente en la pantalla.

### ***Uso de un mensaje con el comando ECHO***

El comando ECHO mensaje es útil si ECHO está desactivado. Para presentar un mensaje que consista de varias líneas sin presentar otros comandos, podrá incluir varios comandos de mensaje ECHO después del comando ECHO OFF en su programa de procesamiento por lotes.

### ***Ocultar el símbolo del sistema***

Si se utiliza el comando ECHO OFF en la línea de comandos, el símbolo del sistema no aparecerá en la pantalla. Para volver a presentar el símbolo del sistema, escriba ECHO ON.

### ***Impedir que MS-DOS presente una línea***

Para impedir que MS-DOS presente en la pantalla una línea determinada en un programa de procesamiento por lotes, inserte el símbolo @ delante del comando.

### ***Presentación de una línea en blanco***

Para mostrar una línea en blanco en la pantalla, escriba ECHO seguido de un punto (ECHO.) sin espacio entre el comando y el punto.

### ***Presentación de caracteres de canalización y de redirección***

No es posible presentar los caracteres de canalización (|) o de redirección (< o >) utilizando el comando ECHO.

El ejemplo siguiente muestra un programa de procesamiento por lotes que incluye un mensaje de tres líneas seguido de una línea en blanco:

```
echo off
echo.
echo Este programa de procesamiento por lotes
echo da formato y verifica
echo nuevos disquetes
echo.
```

Si desea desactivar el comando ECHO y no desea que se presente el comando ECHO en sí, incluya el símbolo @ antes del comando, como en el ejemplo siguiente:

```
@echo off
```

Podrá utilizar los comandos IF y ECHO en la misma línea, como se muestra a continuación:

```
if exist *.inf echo El informe ha llegado.
```

### **FOR**

Ejecuta un comando especificado para cada archivo dentro de un grupo de archivos. Podrá utilizar este comando dentro de un programa de procesamiento por lotes o directamente desde el símbolo del sistema.

Para usar FOR en un programa de procesamiento por lotes, utilice la siguiente sintaxis:

```
FOR %%variable IN (conjunto) DO comando [parámetros-del-comando]
```

Para usar FOR desde el símbolo del sistema, utilice la siguiente sintaxis:

```
FOR %variable IN (conjunto) DO comando [parámetros-del-comando]
```

### **Parámetros**

#### **%%variable o %variable**

Representa una variable reemplazable. El comando FOR reemplazará %%variable (o %variable) con cada cadena de texto en el conjunto especificado hasta que el comando (especificado en el parámetro comando) haya procesado todos los archivos. Use %%variable para ejecutar el comando FOR dentro de un programa de procesamiento por lotes. Use %variable para ejecutar el comando FOR desde el símbolo del sistema.

#### **(conjunto)**

Especifica uno o más archivos o cadenas de texto que serán procesados con el comando especificado. Se requiere el uso de los paréntesis.

#### **comando**

Especifica el comando que será aplicado a cada archivo incluido en el grupo indicado.

#### **parámetros-del-comando**

Especifica cualquier parámetro o modificador que desee utilizar con el comando especificado (si el comando especificado utiliza parámetros o modificadores).

### **Uso de las palabras clave IN y DO**

IN y DO no son parámetros, pero son requeridos por el comando FOR. Si omite cualquiera de estas palabras clave, MS-DOS presentará un mensaje de error.

### **Uso de la variable reemplazable**

Para evitar confusión con los parámetros de procesamiento por lotes %0 a %9, podrá utilizar cualquier carácter para variable, excepto los números del 0 al 9. Para programas de procesamiento por lotes, un solo carácter tal como %%F podría ser suficiente.

Podrá usar valores múltiples para variable en programas de procesamiento por lotes complejos a fin de distinguir entre las distintas variables reemplazables. Sin embargo no podrá anidar (agregar) múltiples comandos FOR en la misma línea de comandos.

### ***Especificación de un grupo de archivos***

El parámetro conjunto puede representar un solo grupo de archivos o varios grupos de archivos. Se pueden utilizar los comodines (\* y ?) para especificar un conjunto de archivos.

Los siguientes son conjuntos de archivos válidos:

**(\**.doc*)**  
**(\**.doc \*.txt \*.me*)**  
**(*ene\*.doc ene\*.inf feb\*.doc feb\*.inf*)**  
**(*ar??1991*)**

Cuando se use el comando FOR, el primer valor en conjunto reemplazará %%variable (o %variable) y MS-DOS ejecutará el comando especificado a fin de poder procesar este valor. Esta operación continuará hasta que MS-DOS haya procesado todos los archivos (o grupos de archivos) que corresponden al valor (o los valores) en conjunto.

### ***Ejemplos***

Supongamos que desea usar el comando TYPE para presentar el contenido de todos los archivos del directorio actual que tengan la extensión .DOC o .TXT. Para hacer esto y usar la variable reemplazable %F, escriba el siguiente comando a continuación del símbolo del sistema:

**for %f in (\*.doc \*.txt) do type %f**

En este ejemplo, todos los archivos que tienen la extensión .DOC o .TXT en el directorio actual serán sustituidos por la variable %F, hasta que se haya presentado el contenido de todos los archivos. Para utilizar este comando en un programa de procesamiento por lotes, simplemente reemplace todas las instancias de %F con %%F. De lo contrario, MS-DOS hará caso omiso de la variable y presentará un mensaje de error.

MS-DOS acepta los modificadores de comandos, símbolos de canalización y símbolos de redirección que desee utilizar con el comando especificado. Por ejemplo, para redirigir la información de salida en el ejemplo anterior a PRN (el puerto de impresora predeterminado), escriba el siguiente comando:

**for %f in (\*.doc \*.txt) do type %f > prn:**

### ***GOTO***

Dirige a MS-DOS hacia una línea marcada por una etiqueta especificada por usted dentro de un programa de procesamiento por lotes. Podrá utilizar este comando sólo en programas de procesamiento por lotes.

El comando GOTO indica a MS-DOS que vaya a una línea específica, identificada por una etiqueta, dentro de un programa de procesamiento por lotes. Al encontrar la etiqueta, MS-DOS procesa los comandos a partir de la línea siguiente.

<b><i>GOTO etiqueta</i></b>
-----------------------------

***Parámetros***  
***etiqueta***

Especifica la línea de un programa de procesamiento por lotes a la que debe dirigirse MS-DOS.

### ***Valores válidos para la etiqueta***

El parámetro etiqueta puede incluir espacios pero no puede incluir ningún otro separador, como por ejemplo, puntos, comas o signos igual. GOTO usa los primeros ocho caracteres de cada etiqueta

El comando GOTO sólo usa los primeros ocho caracteres de una etiqueta. Por lo tanto, las etiquetas "saludos01" y "saludos02" equivalen ambas a "saludos0".

El parámetro de la etiqueta deberá coincidir con la etiqueta del programa de procesamiento por lotes

El valor de la etiqueta que especifique en la línea del comando GOTO deberá coincidir con una etiqueta existente en el programa de procesamiento por lotes. En el programa de procesamiento por lotes, la etiqueta deberá estar precedida por dos puntos (:).

Si el programa de procesamiento por lotes no contiene la etiqueta especificada, dicho programa se suspenderá y MS-DOS presentará el siguiente mensaje:

**La etiqueta no se encuentra**

MS-DOS reconoce una línea de un programa de procesamiento por lotes que comience con dos puntos (:) como una etiqueta y no la procesa como un comando. Si una línea comienza con dos puntos, MS-DOS ignora todos los comandos en ella.

### ***Uso de GOTO para operaciones condicionales***

GOTO se utiliza a menudo con otros comandos en la misma línea de comandos a fin de ejecutar operaciones condicionales. (IF)

El siguiente programa de procesamiento por lotes dará formato a un disquete en la unidad A como un disquete del sistema. Si la operación tiene éxito, el comando GOTO dirigirá MS-DOS a una etiqueta denominada "fin".

```
echo off  
format a: /s  
if not errorlevel 1 goto fin  
echo Ha ocurrido un error durante el formato.  
:fin  
echo Fin del programa de procesamiento por lotes.
```

### ***IF (CONDICION)***

Ejecuta un procesamiento condicional en programas de procesamiento por lotes. Si la condición especificada por un comando IF es verdadera, MS-DOS ejecutará el comando que siga a la condición. Si la condición es falsa, MS-DOS hará caso omiso del comando. Podrá utilizar este comando sólo en programas de procesamiento por lotes.

<b><i>IF [NOT] ERRORLEVEL número comando</i></b>
--

<b><i>IF [NOT] cadena1==cadena2 comando</i></b>
---

<b><i>IF [NOT] EXIST nombrearchivo comando</i></b>
--

### ***Parámetros***

**NOT**

Especifica que MS-DOS deberá ejecutar el comando solamente si la condición es falsa.

### **ERRORLEVEL número**

Especifica una condición verdadera sólo si el programa anterior ejecutado por COMMAND.COM devolvió un código de salida igual o mayor que número.

### **comando**

Especifica el comando que MS-DOS deberá ejecutar si la condición previa se cumplió.

### **cadena1==cadena2**

Especifica una condición verdadera solamente si cadena1 y cadena2 son iguales. Estos valores pueden ser cadenas literales, variables de procesamiento por lotes (por ejemplo, %1) o el valor de las variables de entorno (por ejemplo, PATH). Las cadenas literales no necesitan comillas.

### **EXIST nombrearchivo**

Especifica una condición verdadera si nombrearchivo existe.

### **Ejemplos**

El ejemplo siguiente se utiliza para comprobar la existencia de un directorio determinado. El comando IF no puede utilizarse para averiguar directamente la existencia de un directorio, pero el dispositivo null (NUL) sí existe en cada directorio del disco duro. Por lo tanto, podrá comprobar la existencia de dicho dispositivo para averiguar si existe un directorio en el disco duro.

```
if exist c:\midir\nul goto proceso
```

El ejemplo siguiente presentará el mensaje "No se encuentra el archivo de datos" si MS-DOS no puede encontrar el archivo PRODUCTO.DAT:

```
if not exist producto.dat echo No se encuentra el archivo de datos
```

Cuando un programa se detiene, éste devuelve un código de salida a MS-DOS.

Por ejemplo, generalmente se usa un valor de 0 para indicar que un programa se ha ejecutado con éxito. El parámetro ERRORLEVEL le permite usar códigos de salida como condiciones.

El ejemplo siguiente presentará un mensaje si ocurre un error al dar formato a un disquete en la unidad A. Si no se produce un error, el mensaje no aparecerá.

```
:comenzar  
echo off  
format a: /s  
if not errorlevel 1 goto fin  
echo Ocurrió un error durante el formato.  
:fin  
echo Fin del programa de procesamiento por lotes.
```

Para ver otro ejemplo de cómo se utiliza el parámetro ERRORLEVEL, consulte el comando CHOICE.

### **PAUSE (Pausa)**

Suspende la ejecución de un programa de procesamiento por lotes y muestra un mensaje indicando al usuario que presione cualquier tecla para continuar. Sólo podrá usar este comando dentro de programas de procesamiento por lotes.

<b>PAUSE</b>
--------------

### ***Mensaje al usuario para continuar el programa***

MS-DOS presentará el siguiente mensaje como respuesta al comando PAUSE:

**Presione cualquier tecla para continuar . . .**

Si presiona CTRL+C para detener el programa de procesamiento por lotes, MS-DOS presentará el siguiente mensaje:

**¿Terminar el proceso por lotes? (S/N)**

Si presiona S (Sí) en respuesta a este mensaje, se suspenderá la ejecución del programa de procesamiento por lotes y el control volverá al sistema operativo. Por lo tanto, si hay una sección de un programa de procesamiento por lotes que no desee ejecutar, podrá insertar un comando PAUSE delante de dicha sección. Mientras el programa está detenido a causa de un comando PAUSE, podrá presionar CTRL+C y luego S para suspender la ejecución del programa de procesamiento por lotes.

Supongamos que desea que un programa de procesamiento por lotes pida al usuario que cambie el disquete en una de las unidades de disco. Para ello podría crear el siguiente archivo:

```
@echo off
:inicio
copy a:*. * c:
echo Por favor inserte un nuevo disquete en la unidad A
pause
goto inicio
```

En este ejemplo, todos los archivos del disquete de la unidad A serán copiados en el directorio actual. Después del mensaje solicitando que se inserte otro disquete en la unidad A, el comando PAUSE suspenderá el proceso para permitir que cambie de disquete y luego presione cualquier tecla para reanudarlo. El comando GOTO INICIO enviará al intérprete de comandos a la etiqueta de inicio del archivo de procesamiento por lotes. Para suspender la ejecución de este programa de procesamiento por lotes, presione CTRL+C y después S.

### ***REM (Comentario)***

Permite que se incluyan comentarios (observaciones) o impide la ejecución de comandos en un programa de procesamiento por lotes. (Podrá usar punto y coma (;) para inhabilitar comandos.)

MS-DOS hace caso omiso de cualquier comando o línea de CONFIG.SYS que comience con REM o con un punto y coma (;).

<b><i>REM [comentario] ;REM [comentario]</i></b>
--

### ***Parámetros***

#### **comentario**

Especifica cualquier cadena de caracteres que desee incluir como comentario.

### ***Uso del comando ECHO para presentar comentarios***

El comando REM no presenta los comentarios en la pantalla. Para presentar comentarios en la pantalla, deberá utilizar el comando ECHO ON en el programa de procesamiento por lotes o en su archivo CONFIG.SYS.

### ***Restricciones para uso de caracteres en archivos de procesamiento por lotes***

Los caracteres de redirección (> o <) o el símbolo de canalización (|) no se pueden utilizar en un comentario dentro de un archivo de procesamiento por lotes.

### ***Uso del comando REM para agregar espacio vertical***

Se puede utilizar REM sin ningún comentario para agregar espacios verticales a un archivo de procesamiento por lotes, aunque también se pueden utilizar líneas en blanco. MS-DOS hará caso omiso de las líneas en blanco al ejecutar un programa de procesamiento por lotes.

El siguiente ejemplo muestra un archivo de procesamiento por lotes que utiliza observaciones tanto para explicaciones como para insertar espacios verticales:

```
@echo off  
rem Este programa de procesamiento por lotes da formato y comprueba  
rem nuevos disquetes.  
rem Se denomina NUEVOS.BAT.  
rem  
echo Inserte un nuevo disquete en la unidad B.  
pause  
format b: /v  
chkdsk b:
```

Supongamos que desea incluir una explicación o un comentario en su archivo CONFIG.SYS antes del comando COUNTRY. Para ello, agregue las siguientes líneas al archivo CONFIG.SYS:

```
rem Establecer el código del país como Francia  
country=033
```

El siguiente ejemplo muestra un comando DEVICE que ha sido inhabilitado mediante el uso de un punto y coma (;) en lugar del comando REM:

```
;device=c:\dos\ramdrive.sys
```

### **SHIFT (Cambiar)**

Cambia la posición de parámetros reemplazables en un programa de procesamiento por lotes.

<b><i>SHIFT</i></b>
---------------------

### ***Cómo funciona el comando SHIFT***

El comando SHIFT cambia los valores de los parámetros reemplazables %0 a %9 copiando cada parámetro en el anterior. En otras palabras, el valor de %1 es copiado en %0, el valor de %2 es copiado en %1 y así sucesivamente. Esto es útil para escribir un programa de procesamiento por lotes que realiza la misma operación en cualquier número de parámetros.

### ***Uso de más de 10 parámetros en la línea de comandos***

El comando SHIFT también se puede utilizar para crear un programa de procesamiento por lotes que acepte más de 10 parámetros. Si especifica más de 10 parámetros en la línea de comandos, aquellos que aparezcan después del décimo (%9) serán desplazados uno por uno sobre el parámetro %9.

### ***Desplazamiento de parámetros en sentido inverso***

No hay forma de invertir el comando SHIFT. Una vez ejecutado el comando SHIFT, no podrá recuperar el primer parámetro (%0) que existiera antes de realizarse el desplazamiento.

El siguiente programa de procesamiento por lotes, MICOPIA.BAT, muestra cómo se utiliza el comando SHIFT con cualquier número de parámetros. Copia una lista de archivos a un directorio específico. Los parámetros constan del nombre del directorio seguido de cualquier número de nombres de archivos.

```
rem MICOPIA.BAT copia cualquier número de archivos  
rem a un directorio.  
rem El comando utiliza la siguiente sintaxis:  
rem micopia dir archivo1 archivo2 ...  
set adir=%1  
:obtener_archivo  
shift  
if "%1"==" " goto final  
copy %1 %adir%  
goto obtener_archivo  
:final  
set adir=  
echo Terminado
```