

TÍTULO: *Archivos y carpetas*

OBJETIVOS:

- Entender cómo se organizan los archivos en un ordenador.
- Entender las diferentes formas de almacenar la información.
- Aprender que la extensión de un archivo ayuda a identificar (pero no determina) el tipo de datos en un archivo.

DESARROLLO CONCEPTUAL CONCEPTO/DEFINICIÓN

Los programas de ordenador y los datos que necesitan para trabajar se almacenan permanentemente en archivos informáticos en un disco duro u otro soporte, como una memoria flash, un CD/DVD, o un servidor de Internet. Una misma información se puede guardar en diferentes maneras: depende del uso que se vaya a dar a ese archivo de datos; es decir, quién lo va a leer y escribir, cuál es el tamaño de la información que contiene y/o cómo de rápido se quiere acceder a ella. Esto determina el formato del archivo. A cada archivo se le da un nombre; habitualmente, termina en una extensión: el nombre de un archivo indica su contenido, pero no lo determina.

FORMULACIÓN SIMPLE DEL PROBLEMA

Archivos informáticos y directorios

La RAM de un ordenador es finita y volátil. Sólo caben en ella un número finito de programas y datos. Además, se borran cuando se apaga el ordenador. Para almacenar de forma persistente los programas y los datos se usa el disco duro del ordenador.

El disco duro, como la RAM, guarda la información como una serie de bits (0s y 1s) consecutivos (realmente, guarda polarizaciones magnéticas en la superficie de un disco magnético o polarizaciones eléctricas en elementos de una memoria flash), agrupados en bytes. Las posiciones de estos bytes en el disco se numeran. Para acceder a un grupo de datos que archiva una información, es decir, a un archivo informático, se usan tablas que indican las posiciones en las que comienza cada archivo en el disco: esto es el sistema de archivos de disco (*filesystem*). La estructura de estas tablas (más o menos complicada) buscan tres objetivos: garantizar que se almacenan correctamente los datos (que no se mezclan unos archivos con otros), garantizar que se accede a ellos rápidamente y garantizar que el usuario puede organizarlos de una forma intuitiva. La forma más intuitiva de organizar los datos es en archivos de datos; el análogo cotidiano de esto serían las hojas de papel. La forma más intuitiva de organizar los archivos/hojas de papel sería en carpetas; en

informática se suelen llamar directorios porque, realmente, son archivos que guardan tablas con los nombres que damos a cada archivo/hoja junto con la dirección en el disco en la que se hallan. Las carpetas o directorios se pueden organizar, a su vez, en otras carpetas o directorios. Esto organiza los archivos en una especie de árbol, en el que las ramas salen de los directorios y llegan a otros directorios o terminan en los archivos. Cualquier archivo se ubica en el disco siguiendo una ruta a través de ese árbol desde la raíz del árbol: la ruta absoluta. Un ejemplo (en sistemas Windows) es:

```
C:\Program Files\Mozilla Firefox\firefox.exe
```

Cada \ separa el nombre de un archivo o carpeta del de la carpeta que la contiene (C:\ es la carpeta raíz; se suele dejar el separador al final para indicar que algo es una carpeta).

A veces no se conoce la ruta absoluta, sino sólo la parte final de ella: se conoce una ruta relativa. Por ejemplo, desde la carpeta C:\Program Files\, una ruta relativa al mismo archivo de antes sería

```
Mozilla Firefox\firefox.exe
```

Una ruta relativa puede hacer referencia a la carpeta contenedora. Así, la ruta al archivo

```
C:\Program Files\OpenOffice Beta 4\program\soffice.exe
```

desde la carpeta C:\Program Files\Mozilla Firefox\ sería

```
..\OpenOffice Beta 4\program\soffice.exe
```

donde .. \ indica la carpeta que contiene a aquella en la que estamos. Del mismo modo podemos referirnos a la carpeta actual mediante . \

Formatos de archivo

Básicamente hay dos formas de guardar la información en un archivo informático: en formato binario y en formato de texto.

El formato binario está indicado para que un programa lea rápidamente los datos; se organizan en el archivo de forma parecida a como aparecerán en la RAM durante la ejecución del programa. Los formatos binarios tienen un problema: son específicos del programa que los debe leer; actualmente existen archivos de programas antiguos que ya no se pueden leer porque éstos no funcionan en los sistemas operativos modernos. En la actualidad, los formatos binarios se emplean para almacenar imágenes (como el JPEG de las cámaras fotográficas) y datos científicos (como NC para datos e imágenes meteorológicos); son formatos muy bien documentados y siempre será posible convertirlos sin pérdidas de información a nuevos formatos bien documentados. Otros archivos con formatos binarios son los de las suites ofimáticas antiguas (Microsoft Office, Word Perfect, etc.).

El formato de texto está indicado para que un humano lo pueda leer y escribir con un programa llamado “editor de texto”. El formato de texto sólo contiene un byte por cada letra, número o símbolo que el editor muestra en la pantalla; la equivalencia entre los bytes y los símbolos se llama tabla o código ASCII (American Standard for Computer Information Interchange), por lo que también se habla de formato ASCII. El código ASCII se ha extendido al código UTF (Universal Text Format), pero cualquier archivo ASCII de hace 40 años sigue pudiéndose leer en los ordenadores actuales. En forma de texto se puede codificar toda la información que se quiera, pero de una forma poco eficiente: resultan archivos muy grandes. Por eso leer un archivo de texto ASCII suele ser más costoso para un programa. Sin embargo, la ventaja de no perder datos al cambiar de programa, compensa un mayor tamaño y el tiempo extra de lectura del archivo.

Además de estas dos formas, existe otra más: los archivos comprimidos. Los datos en un archivo pueden presentar redundancia. Por ejemplo, un archivo de texto puede contener la misma palabra repetida varias

veces. Un programa de compresión como el ZIP es capaz de producir un archivo de menor tamaño que el archivo original, que luego se puede descomprimir (con UNZIP) para obtener el original. Hoy en día estos algoritmos de compresión hacen posible guardar información en grandes archivos en formato texto que luego se comprimen para que ocupen un menor espacio en disco y se transmitan más rápidamente por la Red. Es la estrategia detrás de los formatos de los procesadores y hojas de cálculo más modernas.

EJEMPLO

[Este ejemplo depende del sistema operativo del ordenador, es decir, del programa que está ejecutando continuamente el ordenador para leer desde el disco los programas que queremos ejecutar, copiarlos en la memoria e indicarle al procesador donde empezar a ejecutar la primera instrucción de ellos. Dos sistemas operativos muy populares son Windows y Linux, para los que se proporcionan posibles soluciones]

ENUNCIADO

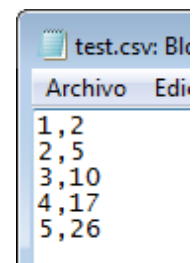
Con un programa de hoja de cálculo, cree una tabla de dos columnas, con números, como si fuesen las coordenadas X e Y de una gráfica. Guárdela en el formato que sugiere por defecto el programa (con extensión XLS, ODS, etc.). Después escoja “Guardar como” y guárdela como archivo CSV (Comma separated values, valores separados por comas). Intente abrir cada uno de los archivos con un editor de texto e inspeccione el contenido.

RESOLUCIÓN

Creemos la tabla en la hoja de cálculo y la guardamos:

	A	B
1	1	2
2	2	5
3	3	10
4	4	17
5	5	26

Nombre	Tipo	Tamaño	Fecha d
test.csv	Hoja de cá...	1 KB	23/04/2
test.ods	Hoja de cá...	7 KB	23/04/2



```
test.csv: Blo...
Archivo Edi...
1,2
2,5
3,10
4,17
5,26
```

Windows. El editor de texto plano de Windows es el Block de notas.

Con él se puede abrir el archivo CSV, pero dará un error al intentar abrir un archivo binario.

Linux. Abriendo un terminal (también llamado emulador de terminal o consola) en el directorio en el que se han guardado los archivos, se puede volcar el contenido de cada archivo a la ventana del terminal usando las órdenes `cat nombre-de-archivo.ext`

(nombre-de-archivo.ext debe sustituirse por el nombre dado al archivo con la extensión correspondiente).

Sólo en un caso, se reconocerá el contenido del archivo: en el CSV.

RESULTADO:

Windows: se utiliza el Administrador de tareas, que se abre haciendo click con el botón derecho del ratón sobre la barra de Windows. Esto abre una ventana en la que se muestran los programas que se están ejecutando en cada momento y el espacio de memoria RAM que ocupan.

Nombre de im...	Nombre ...	CPU	Memoria ...	Descripción
firefox.exe	DRP	12	333.404 KB	Firefox
soffice.bin	DRP	00	53.024 KB	OpenOffice Beta 4.1.0
AcroRd32.exe	DRP	00	48.132 KB	Adobe Reader
MsMpEng.exe	SYSTEM	01	45.732 KB	Antimalware Service E...
explorer.exe	DRP	00	24.032 KB	Explorador de Windows
svchost.exe	SYSTEM	00	19.196 KB	Proceso host para los ...
chrome.exe	DRP	00	18.812 KB	Google Chrome
lyx.exe	DRP	00	17.720 KB	lyx.exe
svchost.exe	SYSTEM	00	7.800 KB	Proceso host para los ...

Linux: se utiliza, en un terminal, la orden

`top`

que muestra los programas que se están ejecutando en ese momento, el porcentaje del tiempo de la CPU que emplean y el espacio que ocupan en la RAM.

La relación entre el espacio ocupado por un programa en la memoria RAM y el tamaño del archivo que lo contiene en el disco depende de qué cantidad de datos requiera el programa para ejecutarse. Por ejemplo, el archivo con el programa de un navegador de Internet ocupará normalmente poco más espacio en RAM que en el disco: el programa suele ser muy grande y las páginas web ocupan poca memoria. Al contrario, un programa de edición de imágenes puede ocupar mucho espacio en RAM cuando está trabajándose con una fotografía de “muchos megapíxeles”.

EJERCICIO DE AUTOCOMPROBACIÓN

ENUNCIADO

Cree con el editor de texto un archivo de texto con pares de coordenadas X Y separadas por punto y coma. Renombre, a continuación, el archivo para que tenga extensión .CSV. Ábralo con un programa de hoja de cálculo.

RESULTADO

Este ejercicio es tan sólo el proceso inverso del ejemplo anterior.

REFERENCIAS:

- Prieto Espinosa, A. Lloris Ruiz, J.C. Torres Cantero. Introducción a la informática. McGraw-Hill, 2006

AUTOR:

- DANIEL RODRÍGUEZ PÉREZ