

Taller 2: Manejo de archivos en C

Pre-taller

1. ¿Que es un archivo?
2. ¿Cual es el propósito de los descriptores de archivos?
3. Investigue al menos 2 funciones para la lectura y escritura de archivos.
4. Investigue la función de los argumentos `int argc` y `char **argv`.
5. El sistema operativo por defecto abre 3 archivos inicialmente, investigue cuales son.
6. Repase el pre-taller 1.

Taller

Este taller se diseñó con la intención de demostrar el uso de las funciones y técnicas en el lenguaje C para trabajar con archivos y pasarle argumentos por la línea de comandos a los programas escritos en este lenguaje. Durante el ejercicio, comente al preparador cada resultado obtenido.

Instrucciones

Compile el código fuente `ejemplo1.c` y ejecútelo de la siguiente forma:

```
./ejemplo1 texto1 texto2 texto3 texto4
```

Este código revela el uso de los parámetros de la función `main argc` y `argv`. Hacen el papel de los parámetros del programa mismo.

Compile el código fuente `ejemplo2.c` y ejecútelo de la siguiente forma:

```
./ejemplo2 entrada.txt
```

Antes de que pueda ser leído o escrito, un archivo tiene que ser abierto usando `fopen`, la función regresa un apuntador que será usado para posteriores lecturas o escrituras en el archivo. La estructura con la que vamos a trabajar es llamada `FILE` y esta definida por `<stdio.h>`.

Los códigos anteriores muestran el uso de la función `fopen` para abrir un archivo y poder trabajar con él; `fgets` para leer el archivo línea a línea; `fclose` para cerrar el archivo (y liberar el recurso); y cómo verificar errores a la hora de trabajar con archivos. Noten la manera en que se envían mensajes personalizados al dispositivo de errores estándar (`stderr`).

También introduce la función `atoi`, que transforma una cadena que representa un número en un valor de tipo `int`. La biblioteca estándar de C ofrece muchas funciones como esta para manipular la información almacenada en cadenas de caracteres.

POST-TALLER

1. Su preparador realizara un quiz de 2 preguntas sobre el taller 1 y 2.
2. Implemente un programa que reciba por parámetros el nombre de un archivo y genere otro que reporte en cada línea el número de vocales que tiene la misma línea en el archivo original.

Ejemplo, suponga el archivo post_taller.txt tiene el siguiente contenido:

```
En un lugar de la Mancha, de cuyo nombre no quiero acordarme, no ha  
mucho tiempo que vivía un hidalgo de los de lanza en astillero,  
adarga antigua, rocín flaco y galgo corredor.
```

El programa (llamémoslo vocales) se invoca desde la línea de comandos de la siguiente forma:

```
./vocales post_taller.txt
```

Él generará un archivo (llamémoslo reporte_vocales.txt) con el siguiente contenido:

```
24  
24  
16
```