

## TAREA 4.

Cruz Martínez Raúl.

Documentación.

**Lenguaje utilizado:** Python, en su versión 3.8.3.

**Entorno de desarrollo:** Este programa fue desarrollado en el IDE Spyder para Windows 10, pero también fue ejecutado a través de la consola del mismo sistema operativo.

**Instrucciones de ejecución:**

Para sistemas operativos Unix, también tendremos que estar situados en el directorio donde tenemos el archivo "tarea4.py" y ejecutar el comando "Python3 tarea4.py". Es necesario destacar que no aseguro una correcta ejecución a través de este método, pues no realicé prueba alguna con él.

**Explicación breve del código:**

El código genera pseudo-archivos como objetos, de modo que estos irán reaccionando, dependiendo de los comandos que introduzca el usuario. Los archivos tienen los métodos: open, close, read, write y seek.

La función "terminal", procesará la entrada dada por el usuario e irá aplicando los métodos a los archivos dependiendo del comando introducido. Además, se encarga de seguir con la ejecución hasta que el usuario introduzca el comando "quit". A través de esta función, podemos crear nuevos archivos dentro del directorio.

Al ejecutar el programa, ya existen precargados tres archivos que se pueden manipular.

**Descripción de los comandos que admite la terminal del programa y su sintaxis:**

Para el uso de la terminal, es necesario que se sigan las instrucciones de esta forma, debido a que el mal uso, puede causar fallos en la ejecución del programa.

- open <arch> <modo>

La instrucción 'open' indicará que trabajaremos con el archivo solicitado. 'arch' hace referencia al nombre del archivo, este archivo debe existir dentro del directorio. 'modo' indica el modo en el que se podrá abrir el archivo.

Esta instrucción devuelve a 'descr', el cual es un descriptor de archivo, el cual nos servirá para referirnos a ese archivo en el resto de la utilización del mismo.

- close <descr>

'close' indica que se finalizará la utilización del archivo. 'descr' es el descriptor que le fue asignado al archivo que quiere cerrar.

- `read <descr> <longitud>`  
 'read' indica que queremos leer un fragmento del archivo. 'descr' una vez más es el descriptor de archivo a leer y <longitud> es la cantidad de caracteres que queremos leer del archivo.
- `write <descr> <longitud> <datos>`  
 Con 'write' podremos leer el archivo indicado con 'descr', que es el descriptor del archivo. 'longitud' es un número que indica la cantidad de caracteres que vamos a escribir. 'datos' es la cadena o información que escribiremos en el archivo.
- `seek <descr> <ubicación>`  
 'seek' indica que queremos desplazar la ubicación desde donde vamos a leer o escribir en el archivo 'descr' es el descriptor del archivo en el que vamos a desplazarnos. 'ubicación', es la ubicación a la que se va a desplazar. Es importante decir que las ubicaciones en el archivo comienzan desde la posición cero.
- `crear <nombre>`  
 'crear' indica que deseamos crear un nuevo archivo en el directorio, el cual tendrá como nombre a lo escrito en 'nombre'. Cada vez que se cree un archivo, este estará vacío y no tendrá algún modo de apertura ni descriptor de archivo. Cuando se comience a escribir en él, se hará desde la posición cero.
- `quit`  
 'quit' nos permitirá terminar la ejecución del programa

### Modos de apertura:

Los archivos pueden abrirse únicamente en tres modos:

- Lectura (R): abre el archivo y la única acción que podrá realizarse será leer el archivo. El archivo no puede ser modificado,
- Escritura (W): si se abre un archivo en modo escritura, aunque este haya tenido contenido, este contenido se descarta, su tamaño es cero y únicamente se podrá escribir en él.
- Modificación (A): este modo nos permitirá leer y escribir en el archivo, al abrirse con este modo, se preserva la información que contenga.

A continuación, se mostrará un *pantallazos* de una ejecución:

```
user/tarea4 $ dir
arch1 [ 5 bytes]
otro_mas [ 11 bytes]
arch2 [ 5 bytes]

user/tarea4 $ open arch1 W
Abriendo el archivo: arch1
Archivo abierto ( W ) -> 1

user/tarea4 $ write 1 7 prueba1

user/tarea4 $ read 1 12
El archivo no se encuentra en modo Lectura o Modificación

user/tarea4 $ close 1

user/tarea4 $ crear arch3

user/tarea4 $ open arch3 A
Abriendo el archivo: arch3
Archivo abierto ( A ) -> 2

user/tarea4 $ write 1 7 prueba2
El archivo no se encuentra abierto

user/tarea4 $ write 2 7 prueba2

user/tarea4 $ read 2 7
prueba2

user/tarea4 $ seek 2 2

user/tarea4 $ read 2 4
ueba

user/tarea4 $ close 2

user/tarea4 $ dir
arch1 [ 7 bytes]
otro_mas [ 11 bytes]
arch2 [ 5 bytes]
arch3 [ 7 bytes]

user/tarea4 $ quit
```