

Nombre: Bohorquez Oclica Nancy

Fecha de entrega: 6 de noviembre de 2024

Sistemas Operativos, Grupo: 06

## **PROYECTO 2**

### **Sistema de archivos multihilo**

Lenguaje de Programación: Python.

**Planteamiento de Problema:** realizar un programa que pueda obtener, crear y modificar información en el micro-sistema-de-archivos FiUnamFs

Siguiendo la especificación que aparece en la siguiente sección, tienen que desarrollar un programa que pueda:

1. Listar los contenidos del directorio
2. Copiar uno de los archivos de dentro del FiUnamFS hacia tu sistema
3. Copiar un archivo de tu computadora hacia tu FiUnamFS
4. Eliminar un archivo del FiUnamFS
5. El programa que desarrollen debe contar, por lo menos, dos hilos de ejecución, operando *concurrentemente*, y que se *comuniquen su estado* mediante mecanismos de sincronización.

## **DESARROLLO**

El programa cuenta con 6 funciones principales:

1. `Create_file_system_disk()`: Inicializa el sistema de archivos con una estructura definida, incluyendo el nombre y la versión. Además de crear un espacio para 16 archivos.
2. `Get_existing_files()`: Lee el disco del sistema de archivos para identificar y cargar los archivos existentes en las listas globales.
3. `Insert_bytes (init_byte, limit,byte, word)`: escribe la cadena especificada en el disco en un rango determinado de bytes y se asegura de que la longitud de la cadena sea correcta, de no serlo la completa.
4. `Copy_from_computer_to_disk(route, queue)`: copia el archivo desde la computadora al disco.

5. `Copy_from_disk_to_computer (file_name, queue)`: copia un archivo desde el disco a la computadora.
6. `Delete_file (file_name, queue)`: elimina un archivo el disco.

Además, contiene una interfaz de Usuario `user_interface()` que permite al usuario realizar las siguientes operaciones:

- Listar los archivos almacenados.
- Copiar archivos entre la computadora y el sistema de archivos.
- Eliminar archivos.

Cada operación de archivo (copiar o eliminar) se ejecuta en su propio proceso, lo que permite que el sistema siga siendo activo durante operaciones que pueden tardar mucho tiempo. Esto se realiza mediante el módulo `threading` y una cola (`queue`) para enviar mensajes al usuario cuando la operación se haya completado.