



Universidad Nacional  
Autónoma de México  
Facultad de Ingeniería



## Segundo proyecto: (Micro) sistema de archivos multihilos

Aguirre Córdova Omar Gabriel 421032167  
Martínez Pavón María Guadalupe 318071280

---

### Objetivo:

Desarrollar es un programa que pueda obtener, crear y modificar información en el micro-sistema-de-archivos que desarrollé para la Facultad de Ingeniería, FiUnamFS.

### LENGUAJE DE PROGRAMACIÓN

Se utilizó Python como lenguaje de programación, y se desarrolló en Pycharm en un entorno Windows .

### Funcionamiento :

#### Información inicial

- Función: Muestra la información básica del sistema de archivos fiunamfs.img.
- Detalles: Al hacer clic, se despliega en el área de texto la configuración y detalles importantes del sistema de archivos, como el ID, versión, etiqueta de volumen, tamaño del cluster, total de clusters, y clusters por unidad. Es útil para verificar que el sistema de archivos está en el formato esperado.

#### Listar Archivos

- Función: Muestra el listado de archivos dentro del sistema fiunamfs.img.
- Detalles: Al hacer clic, se ejecuta la función que carga en la interfaz los archivos permitidos en el sistema (README.org, logo.png, mensaje.jpg). Para cada

archivo, se muestra el nombre, tamaño, fecha de creación y última modificación. Esto permite al usuario verificar el contenido actual del sistema de archivos.

### **Copiar Archivo**

- Función: Copia un archivo desde fiunamfs.img a una carpeta en el sistema local.
- Detalles: El usuario debe ingresar el nombre del archivo en el campo de texto y seleccionar la carpeta de destino en su computadora. Si el archivo existe en el sistema de archivos, se copia al destino seleccionado. Esta operación se realiza en un hilo separado para evitar que la interfaz se congele.

### **Copiar a Sistema**

- Función: Copia un archivo desde el sistema local hacia fiunamfs.img.
- Detalles: Al hacer clic, se abre un diálogo para que el usuario seleccione un archivo en su sistema. El archivo se copia a fiunamfs.img siempre que haya espacio disponible. Esta operación también se ejecuta en un hilo separado, y el listado se actualiza después de la copia.

### **Eliminar Archivo**

- Función: Elimina un archivo de fiunamfs.img.
- Detalles: El usuario debe ingresar el nombre del archivo que desea eliminar. Si el archivo existe en el sistema de archivos, se elimina y se libera el espacio que ocupaba. La lista de archivos se actualiza automáticamente después de la eliminación. Esta operación también se ejecuta en un hilo separado.

### **Desfragmentar Sistema**

- Función: Optimiza el sistema de archivos reordenando los archivos de forma contigua.
- Detalles: Mueve los archivos dentro de fiunamfs.img para llenar los espacios vacíos, optimizando el uso del espacio. Esto es útil después de varias operaciones de copia y eliminación, que pueden fragmentar el almacenamiento en el archivo simulado. Al igual que las otras operaciones, la desfragmentación se ejecuta en un hilo separado.

## **FUNCIONES:**

### **FUNCIONES PRINCIPALES**

- **mostrar\_info\_inicial()** : Muestra la información inicial del sistema de archivos fiunamfs.img, incluyendo ID, versión, etiqueta de volumen, tamaño del cluster, total de clusters, y clusters por unidad.
- **listar\_directorio()**: Carga en la lista entradas\_directorio los archivos permitidos (por ejemplo, "README.org", "logo.png", "mensaje.jpg") que se encuentran dentro de fiunamfs.img.

- **listar\_archivos()**: Muestra en la interfaz gráfica la lista de archivos cargados en `entradas_directorio`, incluyendo su nombre, tamaño, fecha de creación y última modificación.
- **copiar\_archivo\_desde\_sistema()**: Copia un archivo desde `fiunamfs.img` hacia una carpeta en el sistema local del usuario. Usa hilos para ejecutar esta operación sin bloquear la interfaz.
- **copiar\_archivo\_a\_sistema()**: Copia un archivo desde el sistema local del usuario hacia `fiunamfs.img`, siempre y cuando haya espacio disponible. También se ejecuta en un hilo separado para mantener la interfaz responsiva.
- **buscar\_espacio\_disponible(tamano\_archivo)**: Busca y calcula el próximo espacio disponible en `fiunamfs.img` que pueda almacenar un archivo de un tamaño específico.
- **borrar\_archivo()**: Elimina un archivo específico de `fiunamfs.img`, libera el espacio que ocupaba, y actualiza la lista de archivos en la interfaz. Se ejecuta en un hilo separado.
- **desfragmentar\_sistema()**: Reorganiza los archivos en `fiunamfs.img` para almacenarlos de manera contigua, eliminando la fragmentación. Ejecuta la desfragmentación en un hilo para no bloquear la interfaz.
- **mostrar\_estado()**: Muestra el estado de las operaciones en tiempo real (como éxito o fallo de la copia o eliminación de archivos) en el área de texto de la interfaz gráfica, usando una cola para recibir mensajes de los hilos.

## **FUNCIONES AUXILARES**

- **leer\_datos(inicio, tamano)**: Lee una cantidad específica de bytes desde una posición dada en `fiunamfs.img`.
- **leer\_datos\_ascii(inicio, tamano)**: Lee datos en formato ASCII desde una posición dada en `fiunamfs.img`.
- **desempaquetar\_dato(inicio, tamano)**: Desempaqueta un valor en formato little endian desde una posición en `fiunamfs.img`, utilizando `struct.unpack`.
- **verificar\_archivo(nombre\_archivo)**: Verifica si un archivo con un nombre específico existe en `entradas_directorio`. Devuelve el índice del archivo y un valor booleano indicando su existencia.

## **CLASE**

**ArchivoDirectorio**: Clase que representa un archivo dentro del sistema de archivos. Contiene atributos como el nombre, tamaño y cluster inicial del archivo, y se utiliza para almacenar y manejar los archivos que están en `fiunamfs.img`.

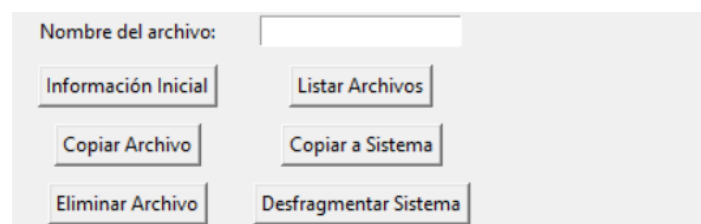
¿Qué tengo que saber / tener / hacer para ejecutar su programa en mi computadora?

## Básico para ejecutar

1. Cómo requisito fundamental se necesita Python 3.x instalado en su computadora.
2. Tener un editor de código o IDE, en nuestro caso utilizamos PyCharm que es muy popular para programar en Python (Puede descargarlo desde [jetbrains.com/pycharm](https://jetbrains.com/pycharm))
3. Este programa utiliza las bibliotecas estándar de Python (os, struct, tkinter, threading, time, queue), que están incluidas en una instalación de Python, así que no necesita instalar dependencias adicionales. OJO: Tkinter: tkinter es necesario para la interfaz gráfica. Viene incluido con Python en la mayoría de los sistemas
4. Guardar el código con extensión .py (puede guardarlo como PROYECTOSO2FINAL.py)
5. Como requisito fundamental se necesita que el diskette esté dentro de la misma carpeta de su archivo y tiene que llamarse "fiunamfs.img", para que pueda leerse correctamente.
6. Lo abre en el IDE
7. Lo ejecuta

## Interfaz gráfica

8. Se le desplegará una interfaz gráfica en donde tendrá 6 botones con las siguientes opciones.



9. El botón de "información inicial", le mostrará la formación general del sistema de archivos
10. Si quisiera visualizar lo que hay dentro del sistema seleccione "Listar archivo"

## Copiar o eliminar archivo

11. Si se quisiera copiar archivo o eliminar un archivo, primero se tendría que poner el nombre del archivo en la barra de arriba y después seleccionar que se debe hacer.
12. Para copiar tendrá que ver en qué carpeta se copiará

## Copiar a sistema

13. Si se desea Copiar a Sistema, se le abrirá la carpeta de archivos para que pueda escoger qué archivo va a querer insertar

## Desfragmentar Sistema

14. Solo presione para desfragmentar y buscar lugar en donde poder insertar archivos.

## Interfaz gráfica

### Información inicial



Gestor de Sistema de Archivos

Nombre del archivo:

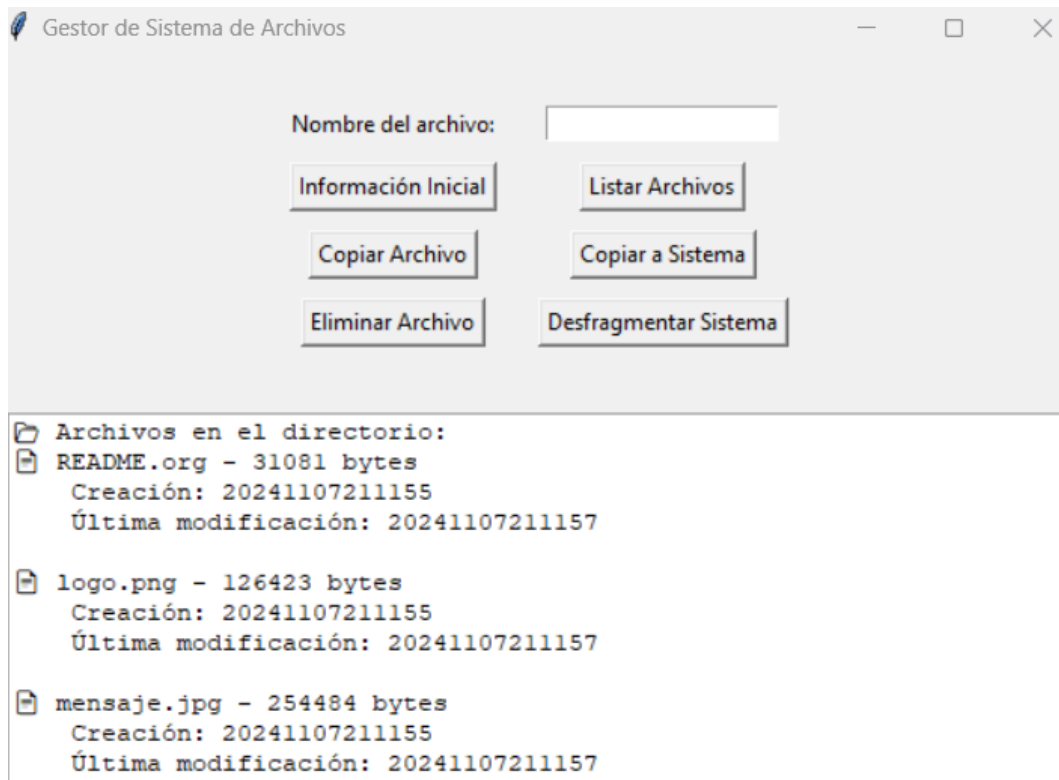
Información Inicial    Listar Archivos

Copiar Archivo    Copiar a Sistema

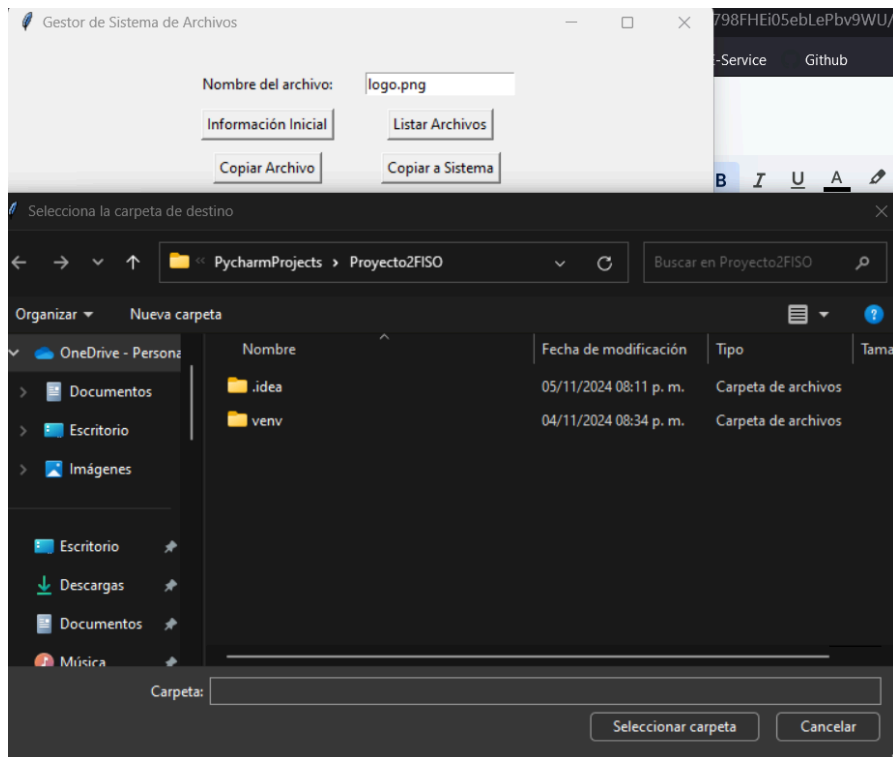
Eliminar Archivo    Desfragmentar Sistema

**Información del Sistema de Archivos:**  
ID del sistema: FiUnamFS  
Versión: 25-1  
Etiqueta de volumen: Un buen sistema  
Tamaño del cluster: 1024 bytes  
Total de clusters: 4  
Clusters por unidad: 1440

## Listar archivos



## Copiar archivo



```
Archivo mensaje.jpg copiado exitosamente a C:/Users/lupit/PycharmProjects/Proyecto2FISO\mensaje.jpg
```

## Eliminar archivo

Nombre del archivo:

Información Inicial

Listar Archivos

Copiar Archivo

Copiar a Sistema

Eliminar Archivo

Desfragmentar Sistema

Archivos en el directorio:

README.org - 31081 bytes  
Creación: 20241107211155  
Última modificación: 20241107211858

mensaje.jpg - 254484 bytes  
Creación: 20241107211155  
Última modificación: 20241107211858

Archivo mensaje.jpg eliminado del sistema de archivos.

## Copiar a sistema

Gestor de Sistema de Archivos

Nombre del archivo:

Información Inicial

Listar Archivos

Copiar Archivo

Copiar a Sistema

Eliminar Archivo

Desfragmentar Sistema

Archivos en el directorio:

README.org - 31081 bytes  
Creación: 20241107211155  
Última modificación: 20241107211858

mensaje.jpg - 254484 bytes  
Creación: 20241107211155  
Última modificación: 20241107211858

El archivo ya existe en el sistema de archivos.

## Copiar a sistema

Nombre del archivo:

Archivos en el directorio:

Archivo copiado exitosamente al sistema de archivos.

## ¿Qué mostró el disquete “fiunamfs.img”?

Encontramos 3 archivos dentro del disquete, de los cuales fueron:

- logo.png



- README.org

```
README.org
Archivo  Editar  Ver

-----
| /Entorno y dependencias/ | Presenta el entorno para el cual el programa fue escrito, detallando según sea pertinente lenguaje
| (incluyendo la implementación y versión mayor), principales módulos que deben ser instalados (con sus respectivas versiones), y demás
| instrucciones pertinentes |
| Indica los principales componentes requeridos para la construcción y ejecución del proyecto, pero omite detalles importantes que
| dificultan su exitosa ejecución |
|
|-----
|
| /Historia en Git/ | El proyecto consta de un mínimo de cinco commits, con información suficientemente descriptiva para comprender
| el proceso de desarrollo
| Consta de un mínimo de cinco commits, pero no presentan un título/comentario suficiente para entender el proceso de desarrollo
| La entrega consta de un sólo commit, no permite entender el proceso de desarrollo del proyecto
| No entregó usando git |
|-----
```



- mensaje.jpg

