



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Facultad de Ingeniería

Micro sistema de archivos multihilos

Proyecto 2

Integrantes:

- *Garcia Riba Emilio*
- *Rivas Gil Maria Lucia*

Profesor: Gunnar Wolf



Objetivo

El objetivo de este proyecto es poder crear un microsistema de archivos multiples donde se pueda:

- Listar los contenidos del directorio
- Copiar uno de los archivos de dentro del FiUnamFS hacia tu sistema
- Copiar un archivo de tu computadora hacia tu FiUnamFS
- Eliminar un archivo del FiUnamFS

Además el programa desarrollado tendrá que contar con mínimo 2 hilos de ejecución operando de manera concurrente y que se comuniquen su estado mediante mecanismos de sincronización.

Desarrollo

Para desarrollar el proyecto y cumplir con los objetivos se implementó el micro sistema de archivos haciendo uso de programación orientada a objetos y la concurrencia con el uso de los hilos.

Se creó una clase llamada : File_system: dentro de esta se incluye toda la lógica relacionada con la lectura y el manejo de la imagen del sistema de archivos. En esta parte se abre el archivo .img y lee el superbloque y así poder obtener la información del nombre del sistema, tamaño de los clusters y el número total de estos.

La lista de objetos “cluster” se crea a partir de la información anterior, esta representa cada uno de los bloques de almacenamiento, también lee la zona del directorio, interpretando cada entrada de 64 bytes y creando objetos “file” que guardan datos como nombre, tamaño, fechas y el cluster de inicio.

Para tener un buen manejo de las operaciones pesadas sin causar una interrupción se hizo uso de hilos. El hilo principal se encarga de mostrar el menú (Aquí se presentan las opciones de acciones que puedes realizar desde el programa, ya sea consultar información del sistema, listar archivos, ver detalles, extraerlos, agregarlos o borrarlos) y aceptar las entradas del usuario, aunado a este se tienen otros dos hilos:

El hilo “mover” se encarga de:

- Extraer archivos

- Agregar archivos
- Eliminar archivos

Utiliza una cola para la recepción de las operaciones, cada operación que implique manipulación de datos en el FiUnamFS se delega a este hilo, haciendo así que el hilo principal permanezca disponible y no realizar tareas pesadas. Al terminar la ejecución se cierran los hilos y el archivo de imagen, para apagarlo de forma ordenada.

Finalmente, para esta implementación se utilizó la programación orientada a objetos como paradigma, ya que permitió organizar la logística del sistema de archivos de forma modular y con mayor claridad. En lugar de que el código fuera completamente lineal y difícil de manejar, se generaron clases para definir las funciones y los datos relacionados, facilitando así el manejo y la estructura general del sistema. Además de esto, la implementación de la concurrencia se integró fácilmente, ya que cada componente puede manejar su propio estado y coordinarse con los demás sin interferencias.

Manual de usuario

Requisitos

- Tener el archivo **fiunamfs.img** en la misma carpeta que el programa.
- Contar con las carpetas:
 1. **input_files/** (para agregar archivos al FS)
 2. **output_files/** (para guardar archivos extraídos)

Ejecución

Para ejecutar el código primero hay que asegurarse de estar en la carpeta que tiene tanto el programa como la imagen y basta con ejecutar en terminal:

python main.py

Uso

Una vez que se ejecuta el programa aparecerá un menú de opciones, dependiendo de lo que se quiera hacer se tecleará el número correspondiente a la acción:

1. Info de sistema
2. Dirs. del sistema
3. Info detallada de un archivo
4. Copiar un archivo hacia nuestro sistema
5. Copiar todos los archivos a nuestro sistema
6. Agregar un archivo a fiunamfs
7. Borrar un archivo de fiunamfs
8. Archivos en input files
9. Archivos en output files
10. Salir

¿Qué hace cada opción?

1. Info de sistema

Muestra información del superblock del FiUnamFS:

- nombre del sistema
- versión
- etiqueta del volumen
- tamaño del cluster
- espacio disponible y ocupado

2. Directorio del sistema

Muestra una tabla con todos los archivos dentro del FiUnamFS:

- nombre
- tipo de archivo
- tamaño
- cluster inicial
- fecha de creación y modificación

3. Información detallada de un archivo

Solicita el nombre del archivo y muestra:

- tamaño en bytes y KB
- fragmentación interna
- clusters que ocupa

- rango de clusters
- eficiencia de almacenamiento

4. Copiar un archivo hacia tu computadora

Permite extraer un archivo desde el FiUnamFS y guardarlo en output_files/.

Solo se ingresa el nombre del archivo cuando se solicita.

5. Copiar todos los archivos

Extrae todos los archivos del FiUnamFS y los guarda en output_files/ automáticamente.

6. Agregar un archivo al FiUnamFS

1. Coloca el archivo que quieras agregar dentro de input_files/.

2. Selecciona la opción 6.

3. Ingresa el nombre del archivo.

4. El sistema revisará:

- si existe espacio suficiente,
- si hay clusters contiguos disponibles.
- y si el nombre no existe ya.

5. Si las condiciones son adecuadas lo copiará dentro del FS y actualizará el directorio.

7. Borrar un archivo del FiUnamFS

Pide el nombre del archivo y lo elimina de la tabla del directorio.

8. Archivos en input_files

Muestra los archivos disponibles para agregarse al sistema.

9. Archivos en output_files

Muestra los archivos que ya han sido extraídos desde el FiUnamFS.

10. Salir

- Cierra el sistema
- Detiene el hilo Mover
- Cierra el archivo fiunamfs.img
- Libera recursos

Consideraciones

- Los archivos que se quieran agregar deben estar primero en input_files/.
- No cerrar la terminal mientras se ejecutan operaciones.
- Evitar nombres de archivo mayores a 14 caracteres.
- Si llega a salir un mensaje como “*La operación ya tomó mucho tiempo*”, puede deberse a saturación del hilo trabajador; solo reintenta.