



**UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO**



Facultad de Ingeniería

Ingeniería en Computación

PROYECTO 3

Alumno: Puente Estrada Rodrigo Abdiel

Profesor: Gunnar Eyal Wolf Iszaevich

Materia: Sistemas Operativos

Semestre: 2024-1

Fecha de entrega: 24 – Noviembre – 2023

Explicación del Programa para Manejar FiUnamFS

Este programa en Python está diseñado para manejar un sistema de archivos ficticio llamado "FiUnamFS". A continuación, se proporciona una explicación detallada de cada parte del código:

Constantes y Variables Globales

```
cluster_size = 1024
directory_start = 1
directory_cluster_size = 4
```

Estas variables representan el tamaño de los clústeres, el inicio del directorio en el sistema de archivos y el tamaño del directorio en clústeres, respectivamente.

Función `list_directory`

```
def list_directory():
    try:
        # Abrir el archivo FiUnamFS en modo lectura binaria
        with open('FiUnamFS', 'rb') as file:
            # Ir a la posición del directorio en el archivo
            file.seek(cluster_size * directory_start)
            # Leer el tamaño del directorio en clústeres
            data = file.read(directory_cluster_size * cluster_size)

            # Iterar sobre las entradas del directorio
            for i in range(0, len(data), 64):
                # Desempaquetar la información de la entrada del directorio
                entry = struct.unpack("<c15sI14s14s14s", data[i:i+64])

                # Extraer información específica de la entrada
                file_type = entry[0].decode('utf-8')
                file_name = entry[1].decode('utf-8').rstrip('\x00')
                file_size = entry[2]
                creation_time = datetime.datetime.strptime(entry[3].decode('utf-8'), '%Y%m%d%H%M%S')
                modification_time = datetime.datetime.strptime(entry[4].decode('utf-8'), '%Y%m%d%H%M%S')

                # Mostrar la información de la entrada del directorio
                if file_type != '/':
                    print(f"Tipo: {file_type}, Nombre: {file_name}, Tamaño: {file_size} bytes")
                    print(f"Fecha de creación: {creation_time}, Fecha de modificación: {modification_time}")
                    print("-----")

    except FileNotFoundError:
```

```
print("El archivo 'FiUnamFS' no se encuentra en la ruta especificada.")
```

Esta función lista el contenido del directorio del sistema de archivos FiUnamFS. Lee las entradas del directorio y muestra información como el tipo de archivo, nombre, tamaño, fecha de creación y fecha de modificación.

Función `copy_from_FiUnamFS`

```
def copy_to_FiUnamFS(file_name):  
    try:  
        with open(file_name, 'rb') as local_file:  
            file_data = local_file.read()  
  
        with open('FiUnamFS', 'r+b') as file_fs:  
            free_cluster = find_free_cluster()  
            if free_cluster is not None:  
                file_fs.seek(cluster_size * free_cluster)  
                file_fs.write(file_data)  
  
            update_directory_entry(file_name, free_cluster, len(file_data))  
  
            print(f"Archivo {file_name} copiado exitosamente a FiUnamFS")  
        else:  
            print("No hay espacio disponible en FiUnamFS para copiar el archivo")  
    except FileNotFoundError:  
        print(f"No se pudo encontrar el archivo {file_name} en tu sistema")  
    except Exception as e:  
        print(f"Ocurrió un error al copiar el archivo a FiUnamFS: {e}")
```

Esta función copia un archivo desde el sistema de archivos FiUnamFS al sistema local. Lee la entrada del directorio para encontrar el archivo, lee sus datos y los guarda en un nuevo archivo local.

Función `copy_to_FiUnamFS`

```
def copy_to_FiUnamFS(file_name):  
    try:  
        # Abrir el archivo local en modo lectura binaria  
        with open(file_name, 'rb') as local_file:
```

```

# Leer los datos del archivo local
file_data = local_file.read()

# Abrir el archivo FiUnamFS en modo lectura y escritura binaria
with open('FiUnamFS', 'r+b') as file_fs:
    # Encontrar un clúster libre en FiUnamFS
    free_cluster = find_free_cluster()
    if free_cluster is not None:
        # Ir a la posición del clúster libre en FiUnamFS y escribir los datos
        file_fs.seek(cluster_size * free_cluster)
        file_fs.write(file_data)

    # Actualizar la entrada del directorio con la información del nuevo archivo
    update_directory_entry(file_name, free_cluster, len(file_data))

    print(f"Archivo {file_name} copiado exitosamente a FiUnamFS")
else:
    print("No hay espacio disponible en FiUnamFS para copiar el archivo")
except FileNotFoundError:
    print(f"No se pudo encontrar el archivo {file_name} en tu sistema")
except Exception as e:
    print(f"Ocurrió un error al copiar el archivo a FiUnamFS: {e}")

```

Esta función copia un archivo desde el sistema local a FiUnamFS. Encuentra un clúster libre en FiUnamFS, escribe los datos del archivo y actualiza la entrada del directorio con la información del nuevo archivo.

Función `find_free_cluster`

```

def find_free_cluster():
    with open('FiUnamFS', 'r+b') as file_fs:
        # Ir a la posición del mapa de bits en FiUnamFS
        file_fs.seek(64) # Suponiendo un superbloque de 64 bytes
        bit_map = file_fs.read(cluster_size) # Leer el mapa de bits

        # Iterar sobre el mapa de bits para encontrar un clúster libre
        for i, byte in enumerate(bit_map):
            for j in range(8):
                bit = (byte >> j) & 1
                if bit == 0:
                    # Encontramos un clúster libre y devolvemos su número
                    return i * 8 + j

        # Si no se encuentra ningún clúster libre, devolvemos None
        return None

```

Esta función encuentra un clúster libre en FiUnamFS examinando el mapa de bits.

Estas son solo algunas de las funciones del programa, que también incluye funciones para copiar archivos desde FiUnamFS al sistema, desde el sistema a FiUnamFS, eliminar archivos, liberar clústeres y desfragmentar FiUnamFS. La función principal `main()` proporciona un menú interactivo para realizar estas acciones.

Ejecución

```
abdielp06@HP-Pavilion-Gaming-Laptop-16-a0xxx:~$ /bin/python3 /home/abdielp06/sistop-2024-1/proyectos/3/PuenteAbdiel/programa.py
Seleccione una acción:
1. Listar directorio
2. Copiar archivo desde FiUnamFS al sistema
3. Salir
Ingrese el número de la acción que desea realizar: 1
El archivo 'FiUnamFS' no se encuentra en la ruta especificada.
```

Salida