

# Informe TP3: FileSystem

Se empieza mostrando de qué forma se puede ejecutar el filesystem:

```
Terminal
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs$ ./fisopfs -f prueba
```

También se puede ejecutar de la siguiente forma, especificando como cuarto parámetro el nombre del archivo con extensión .fisopfs en la que se guardara el filesystem, si se ejecuta como en el primer caso el archivo que se creara tendrá el nombre predefinido de “myfs”, en este caso elijo como nombre tp3 para poner un ejemplo.

```
Terminal
debug] fisopfs_getattr(/)
debug] fisopfs_getattr(/)
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs$ ./fisopfs -f prueba tp3
debug] fisopfs_getattr(/.Trash)
```

Ahora voy a mostrar que se pueden realizar todas las funcionalidades que se esperan del filesystem:

```
Terminal
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs$ touch prueba/archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs$ ls prueba
archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs$ cd prueba
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ mkdir dir_1
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ mkdir dir_2
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo dir_1 dir_2
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ cd archivo
bash: cd: archivo: No es un directorio
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ cd dir_1
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba/dir_1$ touch otro_archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba/dir_1$ ls
otro_archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba/dir_1$ cd ..
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo dir_1 dir_2
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ rmdir dir_2
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo dir_1
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$
```

```
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ rmdir dir_1
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ rmdir dir_1
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo
```

```
Terminal
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ echo voy a escribir en un archi
vo algo >> nuevo_file
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo nuevo_file
```

```

agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ echo voy a escribir en un archi
vo algo >> nuevo_file
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo nuevo_file
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ cat nuevo_file
cat: nuevo_file: No existe el archivo o el directorio
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ echo voy a escribir tambien en
mi archivo ya creado >> archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ cat archivo
voy a escribir tambien en mi archivo ya creado
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ echo voy a appendear otra linea
a mi archivo >> archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ cat archivo
voy a escribir tambien en mi archivo ya creado
voy a appendear otra linea a mi archivo
agus@AgusCD:~/Documentos/sistemas_operativos/tp3/sisop_2022b_g28/fisopfs/prueba$ ls
archivo nuevo_file

```

*(aclaración: olvide mostrar algunas, como el “rm” para eliminar archivos pero también esta implementado y funciona, también se me olvido mostrar el hecho de que se pueden ver las estadísticas con stats y mostrar mas ejemplos con . y ..)*

A continuación, se muestra de que forma se estructuraron internamente los bloques de códigos para almacenar el filesystem.

Se uso esta estructura a la que se llamó “Disco”, a continuación una especificación de la misma:

```

typedef struct disco {
    char directorios[DIRECTORIOS][TAM_NOMBRE_DIR];
    char archivos[ARCHIVOS][TAM_NOMBRE_ARCH];
    char contenidos[ARCHIVOS][TAM_ARCHIVOS_CONT];
    char tabla_arch_dir[ARCHIVOS][TAM_NOMBRE_DIR];
    int dir_actual; //Proximo directorio libre
    int ar_actual; //Proximo archivo libre
    int cant_directorios; //Cantidad de directorios
    int cant_archivos; //Cantidad de archivos
    char nombre[TAM_NOMBRE_ARCH];
} disco_t;

```

Se tendrán 4 arrays, o matrices como prefieran pensarse

- Directorios sera una lista de los nombres de directorios.
- Cant\_directorios dice la cantidad de directorios que tenemos
- Dir\_actual indica la posición del próximo espacio libre en el array Directorios
- Archivos será una lista de los nombres de los archivos
- Contenidos será una lista con los contenidos de los archivos
- Tabla\_arch\_dir será una lista que contenga a que directorio hacen referencia cada archivo
- Cant\_archivos dice la cantidad de archivos que tenemos
- Ar\_actual indica la posición del próximo espacio libre en los array de archivos
- Nombre será simplemente el nombre que se eligió para el archivo .fisopfs en el que se aplicara la persistencia, puede ser elegido por entrada estándar y sino tendrá el nombre predefinido de myfs

Se decidió limitar el tamaño del filesystem a 10 directorios, 20 archivos y 500 caracteres como el contenido para cada archivo, todo esto para mantenerlo simple y priorizar la velocidad, aunque se podría aumentar tranquilamente, de superarse el limite en cualquier momento para la creación de un archivo o directorio se devolverá el error ENOMEM.

También es necesario aclarar que lo que contienen, por ejemplo el array archivos y directorios es netamente su NOMBRE, es decir, no guarda el path, sino que lo limpia antes de guardarlo, es decir, si recibe el path “/archivo” solamente guarda archivo, de la misma forma que si recibe el path “dir\_1/archivo”, también se sigue guardando solamente el nombre archivo.

Finalmente se explica de que forma se modelo la persistencia.

Al principio de la llamada se llama a una función que abre el filesystem, si no encuentra ningún filesystem ya creado con el nombre especificado, lo crea, al ser creado el archivo se creara con el siguiente formato:

```
≡ tp3.fsopfs
1  Directorios:
2  0
3  Archivos:
4  0
5
```

Tanto la línea 1 y 3, las que dicen “Directorios:” y “Archivos:” no son para nada necesarias y bien podrían ignorarse, de hecho es lo que hacen al guardarse, pero se decidieron poner para facilitar la legibilidad en el caso de que se quiera leer el filesystem directamente del archivo o incluso modificarlo.

A medida que vayamos creando archivos, directorios, o desmontemos el sistema, este archivo se va a modificar de la siguiente forma:

```
≡ myfs.fsopfs
1  Directorios:
2  2
3  dir_1
4  dir_2
5  Archivos:
6  2
7  archivo:voy a escribir tambien en mi archivo ya creado:/
8  otro_archivo: voy a escribir en un archivo algo:dir_2
```

Directamente debajo de Directorios y Archivos se dice la cantidad de cada uno respectivamente, del lado de los directorios es bastante sencillo, solo contiene los nombres de los directorios.

Por el lado de los archivos contiene un mínimo parseo, esto esta delimitado bajo el formato:

Nombre del archivo:contenido del archivo:directorio en el que se encuentra el archivo

*(aclaración: se puede notar en la imagen, que se el directorio en el que se encuentra el archivo es el base, se guarda una "/", esto para que quede mejor que un simple null u otro símbolo)*

Con este sistema se busco la simplicidad y la legibilidad sobre todas las cosas, tiene aspectos un poco feos, por ejemplo, cuando se da el caso de appendear líneas, que deben venir precedidas con un salto de línea, se lo guarda en el archivo como una tabulación, para luego al momento de imprimirlo volver a convertirlo, pensé en poner otro carácter especial que cumpla este rol pero creí que podía llegar a ser mas común que la tabulación y ser malinterpretado con el contenido.

Este sistema permite que uno tenga varios "discos" a la vez, entendiéndose por discos como estos archivos .fisopfs, entonces se puede dar el caso de que yo abra mi filesystem llamado tp3, le cree 3 archivos, lo guarde, abra otro filesystem con nombre fs, lo use y después al volver a montar el filesystem tp3 todavia tenga esos 3 archivos en perfectas condiciones.