

Virtual File System(VFS)

Description :

Need to implement a **VFS** for the defined memory size.

Below is the list of **VFS**:

1. File system should support fopen, fread, fwrite, fclose, rename and remove functions.
2. Implement a block device interface to divide the input memory into your defined memory blocks. Here memory block size can be of static size.
3. A file in the **VFS** can span across one or more blocks.
4. Each file should contain a Inode metadata, which should have
 - a. File Name
 - b. Track Memory block used
 - c. File size
 - d. File Authorisation details, which defines which user has read/write permission for the file.
5. **VFS** has to keep track of the blocks which have been allocated and those haven't been.

Concept :

File: A logical container of data on a device

Inode: A container of metadata for an associated file on a device.

Block Device: A **block device** is a computer data storage **device** that supports reading and writing data in fixed-size **blocks**.

File System: A **filesystem** is the methods and data structures that an operating system uses to keep track of files on a disk or partition; that is, the way the files are organized on the disk.

Requirements :

1. The file system needs to keep track of used and unused blocks of device.
2. Demonstrate ability to read and write files concurrently without inconsistency.
3. At Least implement fopen, fread, fwrite and fclose.

Expectation :

1. Make sure that you can execute the code and show that in a working state
2. Use an in-memory store for now. No need to use any database to store data
3. Make sure code functionality correct, and edge cases are covered
4. Code should be easily testable
5. Exceptions are raised and handled appropriately.

6. Separation of concerns is addressed.
7. A driver function that can be used to test the functionality.

VFS

- Get bytes for block id
- Get next free block
- Get total free data

FileHandler

- Open file
- Write file
- Read file
- Change owner

Inode:

- Permissions
 - Read allowed users
 - Write allowed users
 - Execution allowed users
- File name
- File size
- List<BlockId>

InodeDAO

- getInode For file name