San Francisco State University

CSC 415 - Summer 2023

File System - MileStone 1

Team Drivers

Github name: rorymcginnis1

Github repository link:

https://github.com/CSC415-2023-Summer/csc415-filesystem-rorymcginnis1.git

Kaung Nay Htet | 922292784

Himal Shrestha | 922399514

Rory McGinnis | 921337245

James Donnelly | 917703805

1. **A dump (use the provided HexDump utility) of the volume file that shows the VCB, FreeSpace, and complete root directory.**

```
0009D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0009E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0009F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

parallels@ubuntu-linux-22-04-desktop:~/Documents/csc415-filesystem-rorymcginnis1
$ Hexdump/hexdump.linuxM1 --start 0 --count 3 SampleVolume
Dumping file SampleVolume, starting at block 0 for 3 blocks:

000000: 43 53 43 2D 34 31 35 20  2D 20 4F 70 65 72 61 74 | CSC-415 - Operat
000010: 69 6E 67 20 53 79 73 74  65 6D 73 20 46 69 6C 65 | ing Systems File
000020: 20 53 79 73 74 65 6D 20  50 61 72 74 69 74 69 6F |  System Partitio
000030: 6E 20 48 65 61 64 65 72  0A 0A 00 00 00 00 00 00 | n Header........
000040: 42 20 74 72 65 62 6F 52  00 96 98 00 00 00 00 00 | B treboR.◆◆.....
000050: 00 02 00 00 00 00 00 00  4B 4C 00 00 00 00 00 00 | ........KL......
000060: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000070: 52 6F 62 65 72 74 20 42  55 6E 74 69 74 6C 65 64 | Robert BUntitled
000080: 0A 0A 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000090: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0000F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
```

```
0001C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0001D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0001E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0001F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000200: 61 1A AF AA 0A 00 00 00  00 00 00 00 00 00 00 00 | a.◆◆............
000210: 69 6E 67 20 53 79 73 74  65 6D 73 20 46 69 6C 65 | ing Systems File
000220: 20 53 79 73 74 65 6D 20  50 61 72 74 69 74 69 6F |  System Partitio
000230: 6E 20 48 65 61 64 65 72  0A 0A 00 00 00 00 00 00 | n Header........
000240: 42 20 74 72 65 62 6F 52  00 96 98 00 00 00 00 00 | B treboR.◆◆.....
000250: 00 02 00 00 00 00 00 00  4B 4C 00 00 00 00 00 00 | ........KL......
000260: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000270: 52 6F 62 65 72 74 20 42  55 6E 74 69 74 6C 65 64 | Robert BUntitled
000280: 0A 0A 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000290: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0002F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
```

```
0003C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0003D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0003E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0003F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000400: 23 00 00 00 28 4C 00 00  00 00 00 00 00 00 00 00 | #...(L..........
000410: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000420: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000430: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000440: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000450: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000460: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000470: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000480: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000490: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
```

```
0004C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0004F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000500: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000510: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000520: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000530: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000540: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000550: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000560: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000570: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000580: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000590: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0005F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

parallels@ubuntu-linux-22-04-desktop:~/Documents/csc415-filesystem-rorymcginnis1
$ X
```

2. A description of the VCB structure

Our VCB structure is populated in fsInit.c, where it takes the below attributes and populates them.

- The signature value of our volume
- TotalBlockssize in our volume
- The block numbers in our volume
- The freespace initialization
- Initialization to the root directory

3. A description of the Free Space structure

The free space structure is similar to a parking lot. Files on the computer need to be stored similar to how you need to park your car. With this in mind we can agree that it would be nice to know where you parked so you can get back to it hence you would like to know where you are storing the files you created. Each extent in this case has a start which is similar to the parking space number and the count is the amount of parking spaces/ blocks in the extent. So to start the initFreeSpace begins at the sixth spot as it sees that the first 5 spaces are used. loadFredSpace is giving the available blocks able to be taken. serializeFreeSpaceMap keeps track of all extents and simplifies it so we can easily see which blocks are used and unused. The allocateBlocks is what determines what section of blocks works for the data you need to store. releaseBlocks is what gives the space that was taken by allocate back to possibly be used again.

4. A description of the Directory system

The initalize_root_directory function calculates the memory required for each entry, and sets the first two entries name's to "." and ".." respectively. It also populates the rest of the data the file location and size, the location is provided from the freeSpace allocation, and the size is calculated in the function. Furthermore the times created, allocated, and modified are all set to the current time, and isaDirectory is set to 1, to indicate that it is in fact a directory.

5. **A table of who worked on which components**

Name	components
Kaung Nay Htet	Write FsInit and VCB block then populate the block, help with initialization of the free space, and the document, and the commenting.
Himal Shrestha	Worked with James to write the free space, coded frameworks for the VCB and RootDirectory code and the framework for the document, as well as assisting with the documentation, commenting and implementation of the code.
Rory McGinnis	Initialized the root directory function, helped with initializing the functions and data into the initFileSystem, also helped with documentation and commenting.
James Donnelly	Worked with Himal to write the free space, helped with implementation and serializing the free space, as well as assisting with the documentation, commenting and implementation of the code.

Several Individuals also contributed in ways that were ineffable, but quite important. All in all everyone contributed a great deal this week.

6. **How did your team work together, how often you met, how did you meet, how did you divide up the tasks.**

As this milestone was only one week long we had to meet quite frequently to get the work done. Over the course of that week, on the back end of completing an assignment, as a group, last Thursday, we met 4 times over the course of the last week. We met over a discord chat, where we collaborated, and when needed could share our screens and work together. We decided to divide the tasks by the sections provided to us

in the milestone 1 document. Kevin took part 1, Determine if you need to format the volume or not, James and Himal took part 3, as that was the most intense part, Initialize the free space, and Rory took part 4 initialize the root directory. Since Rory and Kevin had less work they implemented part 2 initialization.

7. A discussion of what issues you faced and how your team resolved them.

One issue we had was deciding how to start this. Even though you gave steps this was a project which needs a lot of planning which we are not used to. We had a long talk and figured that we needed to plan.

Another issue that we had with this project was how to divide up the work as we did not know how difficult each part would be since we have never done this before. We ended up assigning pieces to do and if we finished without part we helped others with theirs. In addition we checked each other's code though we do not know how helpful this may have been.

We tried implementing a bitmap which was unsuccessful. No matter what we tried to make it work, we could not figure it out. We tried looking over the code over and over, multiple little revisions that we shared with each other but still nothing. So, oftentimes we had to go back to class recorded lecture videos and get some hints as well as lots of research. The other helpful thing was that we were allowed to use the professor's freeSpace header file as reference which gave us some general idea about what our freeSpace should look like. After that we created our version of the freeSpace file called “extents.h” and “extents.c” and gradually we were able to make progress on that.