File System Milestone One

Team: Bug Master

Team member:

Tun-Ni Chiang 921458769 tunni-chiang

Jiasheng Li 916473043 jiasheng-li

Christopher Ling 918266861 dslayer1392

Shixin Wang 918663491 uyguyguy

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

- HexDump for VCB

The following screenshot shows the block for our volume control block.

- 64 bytes for signature, which in our case is 101.
- 4 bytes for the number of blocks, which converts to decimal is 19531.
- 4 bytes for block size, which converts to decimal is 512.
- 4 bytes for the location of VCB, which is at block #0.
- 4 bytes for bitmap location, which is at block #1.
- 4 bytes for root's starting location, which is at block #6.
- 4 bytes for root's size (in units of blocks), which is a total of 11 blocks.
- 4 bytes for free space starting location, which is at block #17.
- 4 bytes for the next free block, which is at block #17.

From the analysis above, we can see that this hexdump is showing the information of our VCB, and all the data has been written onto the disk successfully.

```
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 1 SampleVolume
Dumping file SampleVolume, starting at block 1 for 1 block:

000200: 31 30 31 00 00 00 00 00  00 00 00 00 00 00 00 00 | 101.............
000210: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000220: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000230: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000240: 4B 4C 00 00 00 02 00 00  00 00 00 00 01 00 00 00 | KL..............
000250: 06 00 00 00 0B 00 00 00  11 00 00 00 11 00 00 00 | ................
000260: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000270: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000280: 00 00 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 | ................

000300: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000310: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000320: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000330: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000340: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000350: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000360: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000370: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000380: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000390: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0003A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0003B0: 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 | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
```

- **HexDump for BitMap**

  According to our screenshot, we have hexadecimal starting from FF FF 80 which can convert to binary number 1111 1111 1111 1111 1000. We know block 0 is the VCB, blocks 1-5 are free space management, and blocks 6-17 are root directory. As a result, those 17 bits of the bitmap should be marked as used which is '1' after initializing the file system.

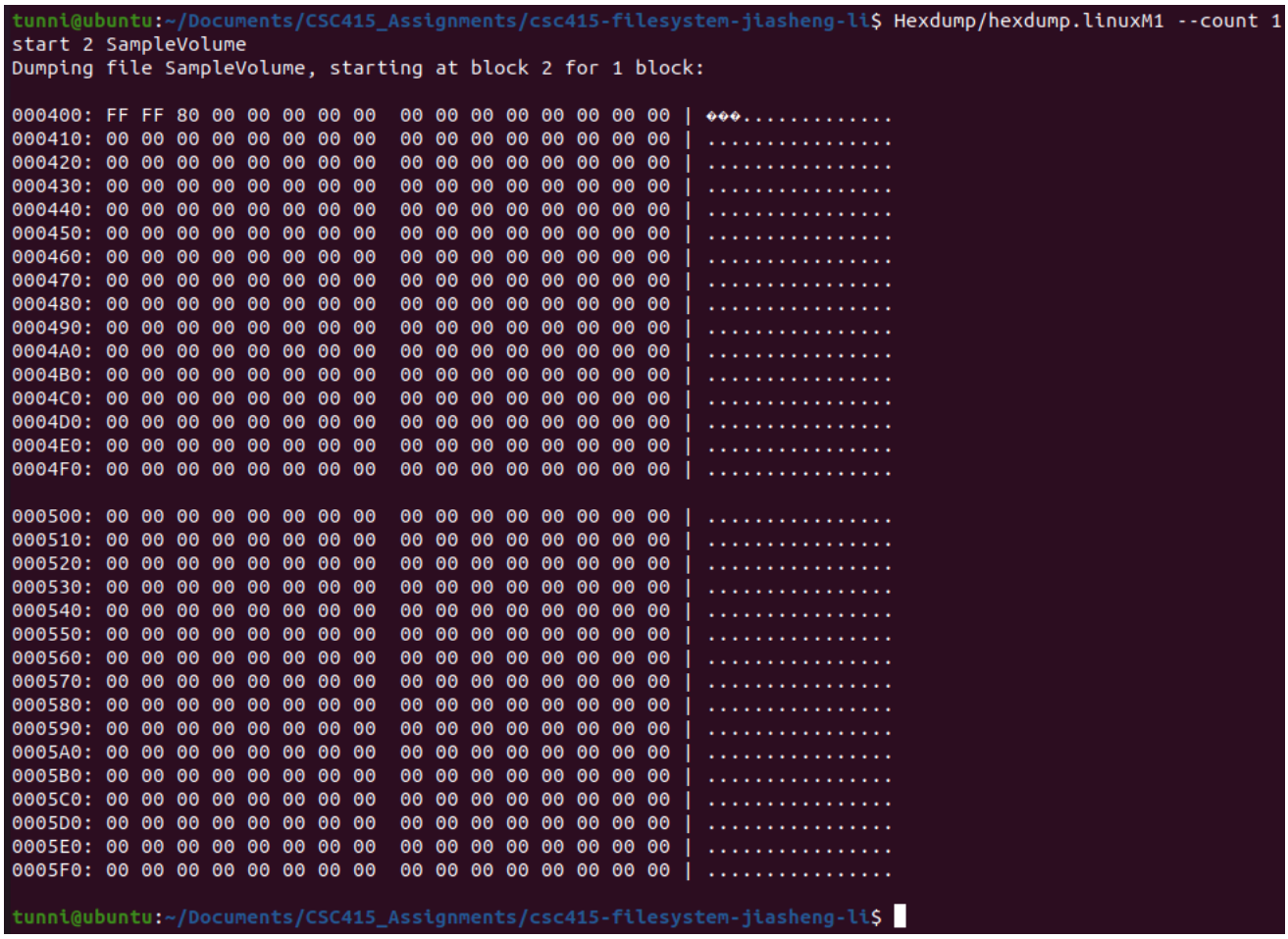
  1111 1111 1111 1111 1000

  VCB

  Free Space

  Root Directory

  Also, there are some bits marked as used at the end of the last block. This is because we malloc more than the actual number of bits. (We did 5 * blockSize = 2560 bytes, which is 20,480 bits. But the total number of blocks is 19531, which we only need 2442 bytes.) Therefore, in order to prevent accidental usage, we marked it as used (equal to 1).



```
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 3 SampleVolume
Dumping file SampleVolume, starting at block 3 for 1 block:

000600: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000610: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000620: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000630: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000640: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000650: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000660: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000670: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000680: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000690: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0006F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000700: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000710: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000720: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000730: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000740: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000750: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000760: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000770: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000780: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000790: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0007F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 4 SampleVolume
Dumping file SampleVolume, starting at block 4 for 1 block:

000800: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000810: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000820: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000830: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000840: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000850: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000860: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000870: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000880: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000890: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008D0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008E0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0008F0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000900: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000910: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000920: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000930: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000940: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000950: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000960: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000970: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000980: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000990: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0009A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0009B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
0009C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
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 | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
```

```
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 5 SampleVolume
Dumping file SampleVolume, starting at block 5 for 1 block:

000A00: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A40: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A50: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A60: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A80: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000A90: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AA0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AB0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AC0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AD0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AE0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000AF0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000B00: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B40: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B50: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B60: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B80: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000B90: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BA0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BB0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BC0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BD0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BE0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000BF0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 6 SampleVolume
Dumping file SampleVolume, starting at block 6 for 1 block:

000C00: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C40: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C50: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C60: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C80: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000C90: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CA0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CB0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CC0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CD0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CE0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000CF0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000D00: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D40: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D50: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D60: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000D80: 00 00 00 00 00 00 00 00  00 1F FF FF FF FF FF FF | .................
000D90: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DA0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DB0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DC0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DD0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DE0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................
000DF0: FF FF FF FF FF FF FF FF  FF FF FF FF FF FF FF FF | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
```

- **HexDump for Root Directory**

  The following screenshot shows the block for our root directory. The size of our directory entry(DE) structure is 104 bytes, and the hexdump is showing the information for the first and second directory entries.

- 64 bytes for names, which "." for first DE and ".." for second DE
- 4 bytes for directory size, which is 5632 bytes. (Although the root contains only 54 DE, since we allocated 11 blocks for it, so it will have the size of 11 * blockSize)
- 4 bytes for pointing location, which in the root cause, first and second DE points to itself, so it points at block #6.
- 8 bytes for isDir, which in the root case, both of the DE are directory because they are root itself.
- 8 bytes for createTime
- 8 bytes for lastModTime
- 8 bytes for lastAccessTime

```
tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$ Hexdump/hexdump.linuxM1 --count 1 --
start 7 SampleVolume
Dumping file SampleVolume, starting at block 7 for 1 block:

000E00: 2E 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E40: 00 16 00 00 06 00 00 00  01 00 00 00 00 00 00 00 | ................
000E50: 7B 8F 77 61 00 00 00 00  7B 8F 77 61 00 00 00 00 | {.wa....{.wa....
000E60: 7B 8F 77 61 00 00 00 00  2E 2E 00 00 00 00 00 00 | {.wa............
000E70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E80: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000E90: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000EA0: 00 00 00 00 00 00 00 00  00 16 00 00 06 00 00 00 | ................
000EB0: 01 00 00 00 00 00 00 00  7B 8F 77 61 00 00 00 00 | ........{.wa....
000EC0: 7B 8F 77 61 00 00 00 00  7B 8F 77 61 00 00 00 00 | {.wa....{.wa....
000ED0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000EE0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000EF0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

000F00: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F10: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F20: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F30: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F40: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F50: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F60: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F70: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F80: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000F90: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FA0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FB0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FC0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FD0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FE0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................
000FF0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................

tunni@ubuntu:~/Documents/CSC415_Assignments/csc415-filesystem-jiasheng-li$
```

2. A description of the VCB structure

Our VBC structure has:

a) The holding signature of our volume
b) The number of blocks in our volume
c) The size of each block in our volume
d) The locations to VBC itself, which locates at block 0
e) The bitMap starting location
f) The root directory starting location
g) The number of blocks that have been allocated for root
h) The free space starting location
i) The location of the next free block

3. A description of the Free Space structure

We use bitmap for our free space management. We malloc the memory which is equal to 5 blocks and multiply 512 bytes of each block to the bitmap. The bitmap holds a bit representing free space by '0' or used space by '1'.After initializing the VCB, we need to mark the first 6 bits of the bitmap to be used since the first block is for VCB and blocks 1- 5 are for bitmap. We also need to mark the remaining bits of 19,531 blocks to be free. Besides, we mark the bits that are over 19, 531 also to be used. The char * array is the array working for byte, but the bitmap is working for bits. As a result, we need to calculate the working byte for the array of the bitmap by using numbers of bits divided by 8. Then, we get the bit location of that working byte by modding numbers of bit by 8 to get the remainder. We got help from Professor Robert by adding setArray and clearArray to help us handle the flipping of a certain bit. Finally, we set bit to be used by implementing array[bitIndex / 8] = array[bitIndex / 8] | setArray[bitIndex % 8] and set bit to be free by implementing array[bitIndex / 8] = array[bitIndex / 8] & clearArray[bitIndex % 8]. We have the following functions to implement our free space management:

a) int initFreeSpace(VCB *vcb, int blockSize) - initializes the bitmap after initializing the VCB
b) reloadFreeSpace(VCB *vcb, int blockSize) - reloads the bitmap in memory and write it into a disk
c) allocateFreeSpace(VCB *vcb, int blockCount) - sets requested bits to be used and update the first free block location of the bitmap
d) releaseFreeSpace()

e) setBitUsed (unsigned char *array, int bitIndex) - sets a certain bit of the bitmap to be use
f) setBitFree(unsigned char *array, int bitIndex) - sets a certain bit of the bitmap to be free
g) checkBitUsed(unsigned char *array, int bitIndex) - checks whether a certain bit of the bitmap to be use

4. A description of the Directory system

Our directory entry structure has:

a) The name of a directory entry which is a char-type array
b) The size of a directory entry
c) The pointing location of a directory entry which is a logic block number
d) The directory entry's creating a timestamp
e) The directory entry's last-modifying timestamp
f) The directory entry's last-accessed timestamp

5. A table of who worked on which components

Components	The Assigned Teammates
Dump	Tun-Ni Chiang, Christopher Ling, Jiasheng Li, Shixin Wang
VCB Structure	Tun-Ni Chiang, Christopher Ling, Jiasheng Li, Shixin Wang
Free Space Structure	Tun-Ni Chiang, Christopher Ling, Jiasheng Li, Shixin Wang
The Directory System	Tun-Ni Chiang, Christopher Ling, Jiasheng Li, Shixin Wang

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

We met every Friday at 11 am on zoom meeting and set up an office hour with Professor Bierman every Monday; otherwise discussed things on Discord. We have our own branch

on our GitHub and built our own code. Discord is the most active place for us to communicate. Whenever we have a question, we will throw it there and other teammates will solve it together.

7. **A discussion of what issues you faced and how your team resolved them.**
   - Initialization of VCB
     a) Problem: How to check the VBC is initialized?

        Team solution: We set up an office hour with Professor Robert. When the VCB has been initialized, it has a signature that is the value that we assigned. Whenever we try to initialize our file system, we need to check the signature of the VCB to decide whether the file system needs to initialize the volume or load up our bitmap.

   - Bitmap structure
     a) Problem: The modification of bitmap by modifying the array of bytes which is incorrect since the bitmap deals with bits

        Team solution: We set up an office hour with Professor Robert. We need to calculate the working byte of the array of the bitmap by using the number of bits divided by 8 and the bit location of that working byte by moding number of bits by 8. We need to define a setArray[8] = {0x80, 0x40, 0x20, 0x10, 0x08, 0x04, 0x02, 0x01} and a clearArray[8] = {7F, BF, DF, EF, F7, FB, FD, EE}. After that, we have set bitmap method Array[x/8] = Array[x/8] | setArray[x%8] and clear bitmap method Array[x/8] = Array[x/8] & setArray[x%8]

   - Initializing the directory field
     a) Problem: The initial value of some data fields of struct Directory Entry such as createTime, lastModTime, and lastAccessTime.

        Team solution: We set up an office hour with Professor Robert. Since these variables are not pointer-type variables that should be initialized NULL, we should initialize them as 0 to be Null state.

   - Location field in DE

     a) Problem: what is the value of the location field of a directory entry?

        Team solution: We reviewed the lecture video and find that the location is what the directory entry wants to point to. On the other hand, the location is the logic block number that the directory entry locates in the memory.

Overall, the meetings and the professor's slack message helped piece things together for milestone 1.