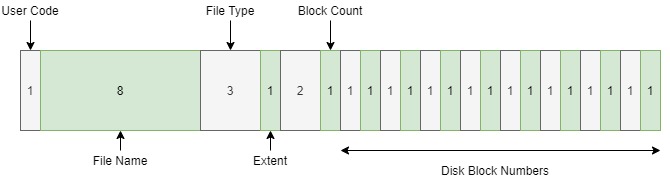
Project 4 Report

1. Design of cpmFS:

The directory entries are structured as follows:



Block 0 contains the directory which contains the pointers to each location in the disk. This disk is 256 blocks, which contain 1024 bytes each. The first byte indicates whether an extent is used or not. The next 8 bytes contains the file name, followed by 3 bytes containing the file extension (e.g. .txt or .py). the next 4 bytes contain both the number of bytes past the last full sector aswell as the number of sectors used. Finally the rest of the bytes contain the data for the file.

To implement such a system, I used a function-oriented approach. I started with the code provided to fill in the stubs. The functions are as follows:

• mkDirStruct()

• writeDirStruct()

• makeFreeList()

• printFreeList()

• cpmDir()

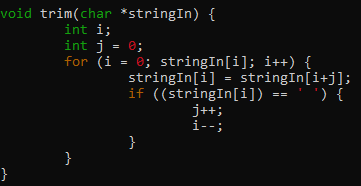
• checkLegalName()

• findExtentWithName()

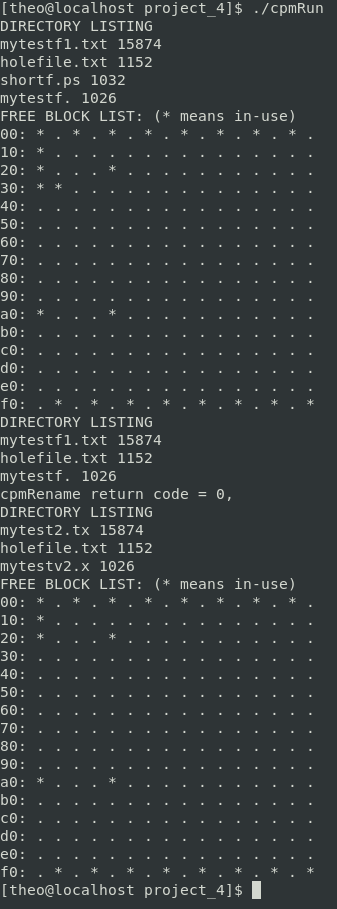
• cpmDelete()

• cpmRename()

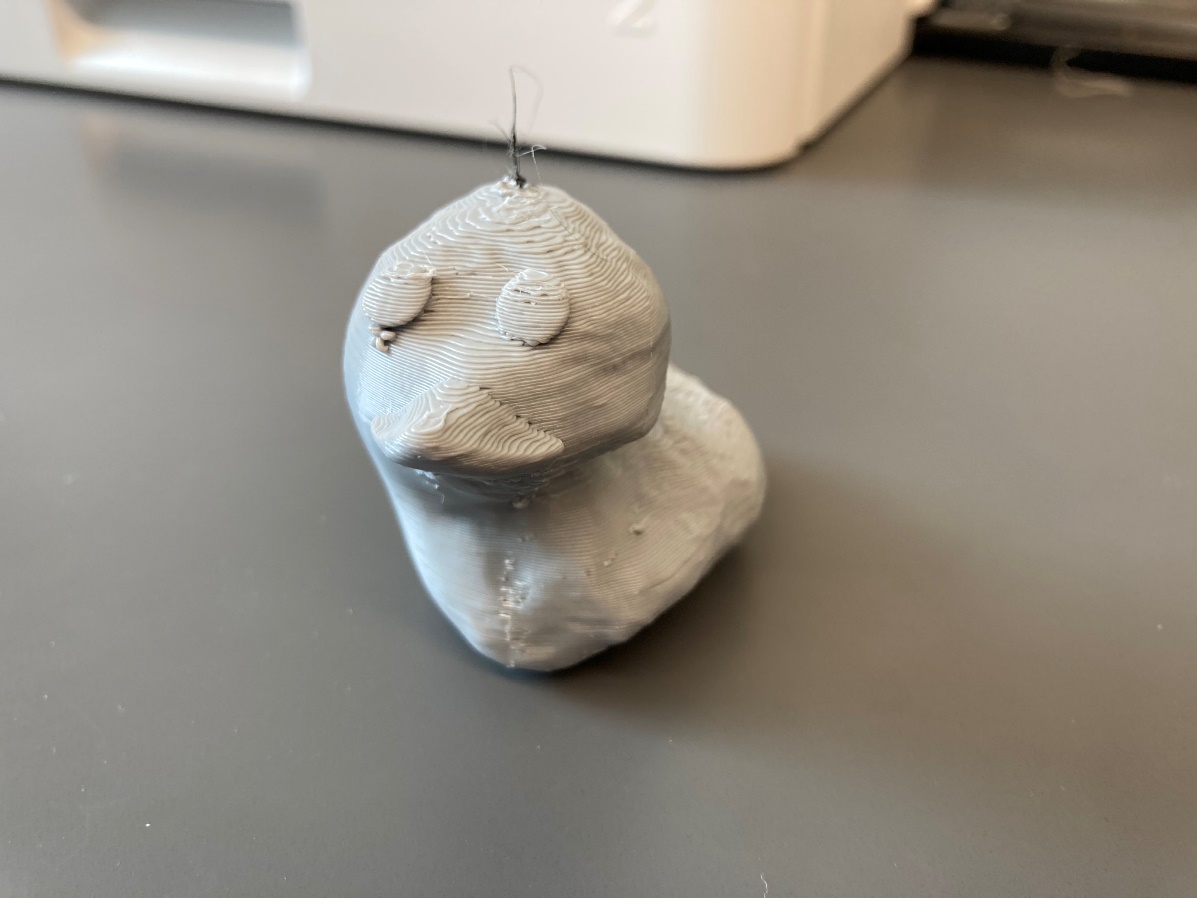
I also implemented a method to trim spaces from string as I found it saved me a lot of code (pictured below).



All functionality was full tested using centOS a sample output can be seen below:



1. Lessons Learned:
2. Start the project on time: I started this project before we went over it in class and thus have gotten it done well before the deadline.
3. Spend time to fix Warning in C: by fixing the warnings I believe I saved myself a lot of debugging as soon as my project was complete I had very few errors that needed to be fixed.
4. Using a rubber duck to talk through my code is very useful. Here is a picture of my duck. His name is Gerald (he is not the smartest, or the best looking, but he gets the job done).



1. References:

I used the following website to look up c syntax:

geeksforgeek.org, cpluscplus.com, stackoverflow.com, and tutorialspoint.com

I also used just about everything Dr. Qin put on canvas related to this project.