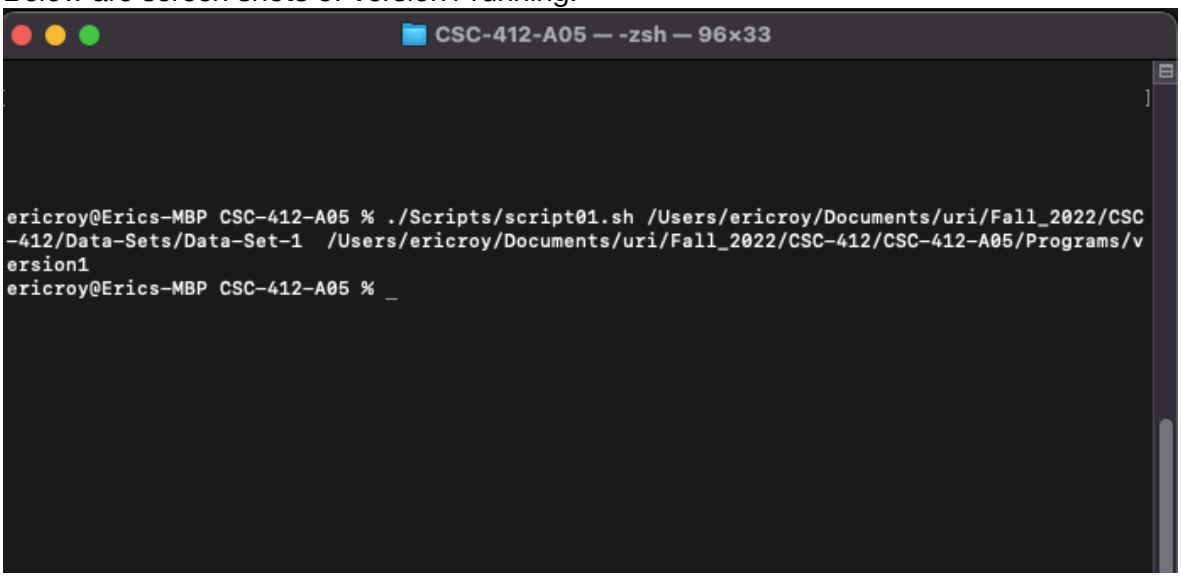
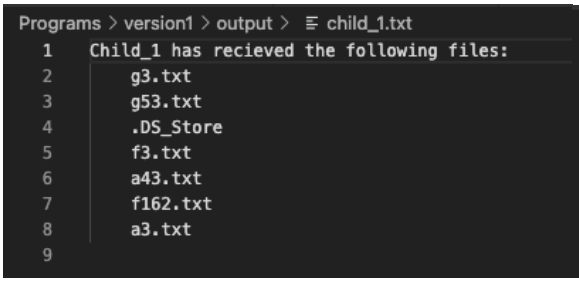
Eric Roy
CSC-412-A05
Program Report

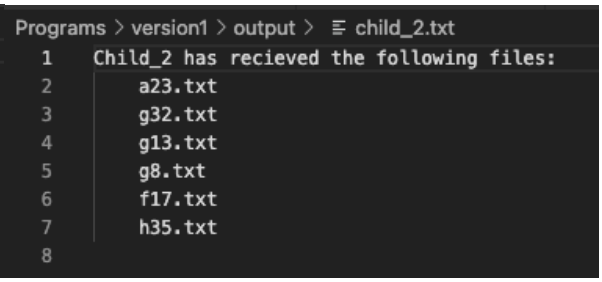
For this assignment I was only able to complete Version 01 and Version 02.

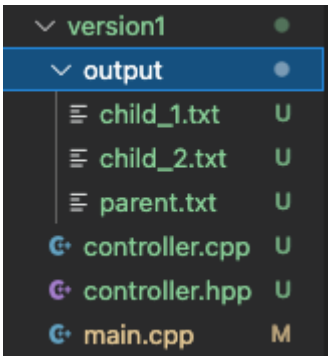
Version 01:

For version one I have a main.cpp, controller.hpp and controller.cpp. The controller files declare and define the functions used within main.cpp. I did this to keep the main.cpp file neat and organized and easy to read. Having a controller.hpp also allowed me to use doxygen comments to better describe the functionality of what each function does. Version 1 is compiled and ran through the command ./Scripts/script01.sh *<DATA DIRECTORY> <SCRAP DIRECTORY>.* Script01.sh will search the data directory and determine how many child processes there will need to be in the program. It is passed to build01.sh and then it is compiled and ran. The output of version01 will be n+1 text files where n is the number of child processes and +1 accounts for the singular parent output file. For Version 1, the child output files contain only the file names it received in the initial split of work. The initial split of work is done by diving the total number of files by the number of expected child processes. For uneven splits the remainder from the division is passed to the first n child processes, where n is the number of extra files that wasn't evenly split between the children. The parent output file does not contain any important information. I simply has the line "This is the parent output file" Below are screen shots of Version1 running.



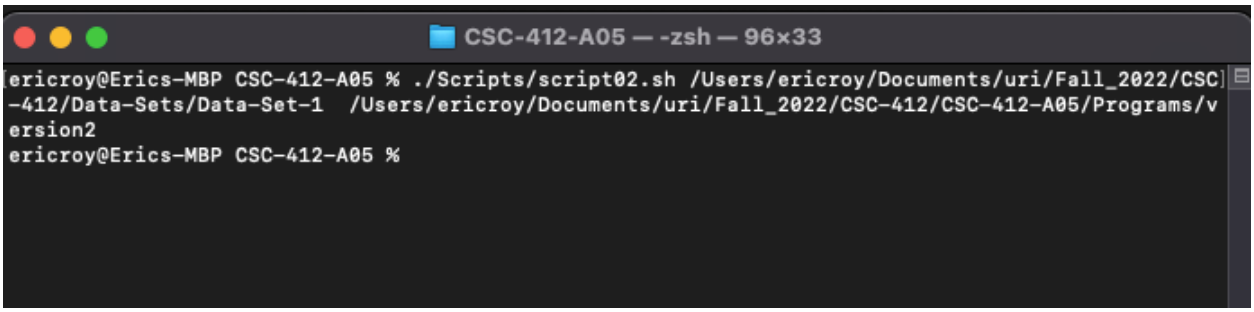


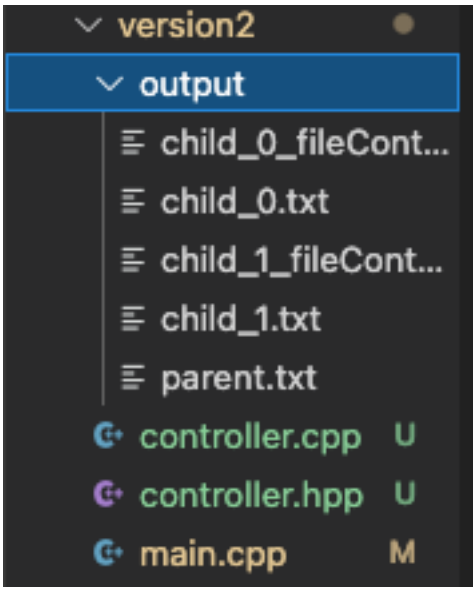


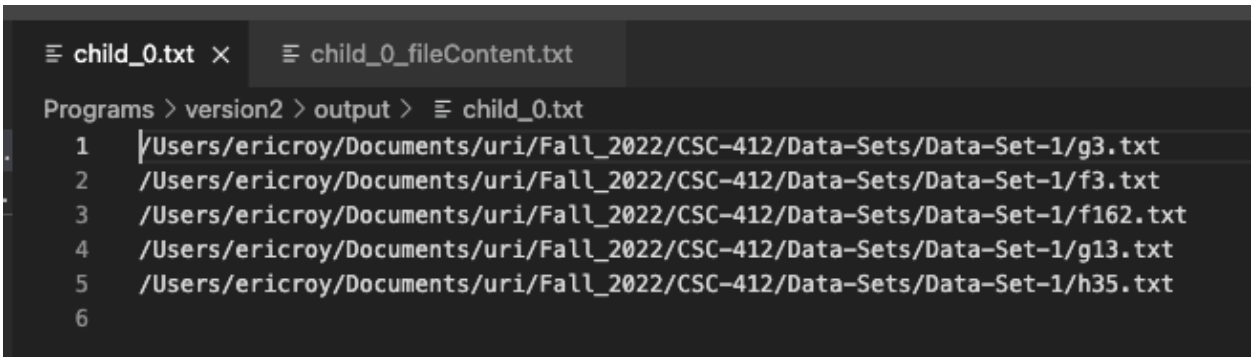


Version 02:

For Version2 the arrangement of files is the same as version1. There is a main.cpp controller.hpp and controller.cpp. Version2 does everything that version1 does however instead of stopping at the initial split of work, each child process opens the files that they receive and verify if the file actually belongs to them. If it does, they store it for processing, if they dont the communicate back to the parent process that the file doesn't belong to them. The parent then takes the file path sent back to it, and passes to the correct child process. This all happens in the first generation of child processes. Before ending, each child will output their verified list of files that they need to process to the output directory. Once each child process ends the parent process then initiates a second generation of children processes and they are tasked with opening the files they have and outputting its contents to a .txt file. The file they output to is child_<number>_fileContent. Once the second generation of child process complete the parent then outputs its own file. This time the content of the parent output file is the combined output of all of the children's fileContent files. Below are screen shots of this version working. For both programs some of my shortcomings are not being more modular with some the larger functions I have. In the beginning when I started this assignment I was doing a decent job keeping the functions small however towards the end of the assignment when getting closer to submission a bad habit is to make a function do many things as if it were a mini main.cpp. Being aware of this my goal for next assignment is to avoid this and focus heavily on modularity in a program.


CSC-412-A05 — -zsh — 96×33
ericroy@Erics-MBP CSC-412-A05 % ./Scripts/script02.sh /Users/ericroy/Documents/uri/Fall_2022/CSC
-412/Data-Sets/Data-Set-1 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/CSC-412-A05/Programs/v
ersion2
ericroy@Erics-MBP CSC-412-A05 %


version2
output
child_0_fileCont...
child_0.txt
child_1_fileCont...
child_1.txt
parent.txt
controller.cpp U
controller.hpp U
main.cpp M


child_0.txt
child_0_fileContent.txt
Programs > version2 > output > child_0.txt
1 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/Data-Sets/Data-Set-1/g3.txt
2 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/Data-Sets/Data-Set-1/f3.txt
3 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/Data-Sets/Data-Set-1/f162.txt
4 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/Data-Sets/Data-Set-1/g13.txt
5 /Users/ericroy/Documents/uri/Fall_2022/CSC-412/Data-Sets/Data-Set-1/h35.txt
6


child_0.txt
child_0_fileContent.txt
Programs > version2 > output > child_0_fileContent.txt
1 0 2 // Created by Jean-Yves Hervé on 2020-10-10.
2 0 4 #include <stdio.h>
3 0 1 // Hello ----- main.c
4 0 5
5 0 3
6

child_0.txt | child_0_fileContent.txt | *parent.txt* ×

Programs > version2 > output > parent.txt

```
1    0 2 //  Created by Jean-Yves Hervé on 2020-10-10.
2    0 4 #include <stdio.h>
3    0 1 //  Hello  -----  main.c
4    0 5
5    0 3
6    1 12
7    1 10    return 0;
8    1 6 int main(int argc, char* argv[])
9    1 11 }
10   1 7 {
11   1 9     printf("CSC412 - Programming assignment 04\n");
12   1 8     printf("Hello, World!\n");
13
```