Project 1 - Basic Data Sorter - multiproc

Fareen Pourmoussavian - 165007750 - fp183
Neel Patel - 163007037 - nap166

Design:

- We noticed we could use most of our code from the previous project. So our oldmain method is the main method from our previous project that handled taking in the .csv file and sorting it and our main is our new main method which recursively deals with traversal and forking.

Compilation:

- Inside zip folder will be mergeSort.c, sorter.h, sorter.c file.
- For sorter.c use $gcc sorter.c -o sorter

Assumptions:

We wrote a rather detailed outline of all the sections of the code, to avoid any surprises. The only surprises were how many extra methods were needed to handle extraneous cases. Mergesort wise, it uses the same code as the one from project 0, so there should be no difference there.

Difficulties:

- PID outputs
    - Unfortunately, even by the deadline we were unable to figure out how to print all the PIDs in the same array, across multiple processes. So the end result is multiple printouts of the printPIDs function, and all the PIDs exist across all the multiple prints. The highest processes value is correct.
    - It seems like there is one printPIDs print for every csv fork.
- Directory traversal
    - Initially we attempted the traversal in a recursive manner, but we soon found out that it was a bad idea, as each spawn would fork as well, leading to unnecessary forks. Therefore we switched to a “linear” form of traversal where forking is done correctly.

Testing Procedure:

To test this, we made test folders with test files and test sub folders to see whether our program would not only sort but also work if the input and output directories were the same and if the output directory was different than the input directory.

	No -d flag	Bad -d path	Valid -d path
No -o flag	Assume current directory for both	Fail gracefully	Assume current directory for -o, use valid directory for -d
Bad -o flag	Fail gracefully	Fail gracefully	Fail gracefully
Valid -o flag	Assume current directory for -d, use valid for -o	Fail gracefully	Use the valid directories