Name: Daivakshi Vaidya **Student Number:** 217761016

EECS 3221 - OS Fundamentals Assignment 1 - Report

1. General outline of the assigned work

process.c	Used system calls to run parallel child programs and piped the results to the main program for further processing. The main program gets command line arguments to determine how many datasets are to be processed and hence how many child processes need to be started, it also gives the filenames. The main process obtains the necessary information, declares variables and then forks into the the clid process. The child process runs in a loop that determines how many forks need to be created. The main program also initialises a pipe that can be used to communicate the sum of all float numbers and their count from the child to parent. The child takes the filename, reads the file and stores elements into an array while also keeping count of the numbers. It then computes the sum of all numbers and writes it in an array along with num (count of numbers). This array is written in the write-end of the pipe. The parent then reads this result from the read-end. Parent waits for all processes to finish executing and receives all sum and num values from child. Then it calculates average for each dataset and a total average of all datasets. The program then prints all the results.
thread.c Using pthreads	The main program thread calls to the runner function (worker thread function) in a loop. Command arguments determine filenames and number of threads to be created. The main tread passes the attributes and parameter to the worker thread (parameters like file name). The worker thread then processes each dataset, calculates sum and counter and then stores the result on shared memory. This shared memory can be accessed by both the main thread and the worker threads. Main thread wait for all threads to finish running and once it has access to sum and num (counter) it calculates average for each dataset as well as the total average. The worker threads finish when it runs into an exit command.

In summary, pthread uses shared memory for parallel execution while processes.c uses pipes to communicate between child and parent.

2. Assigned work/components done successfully

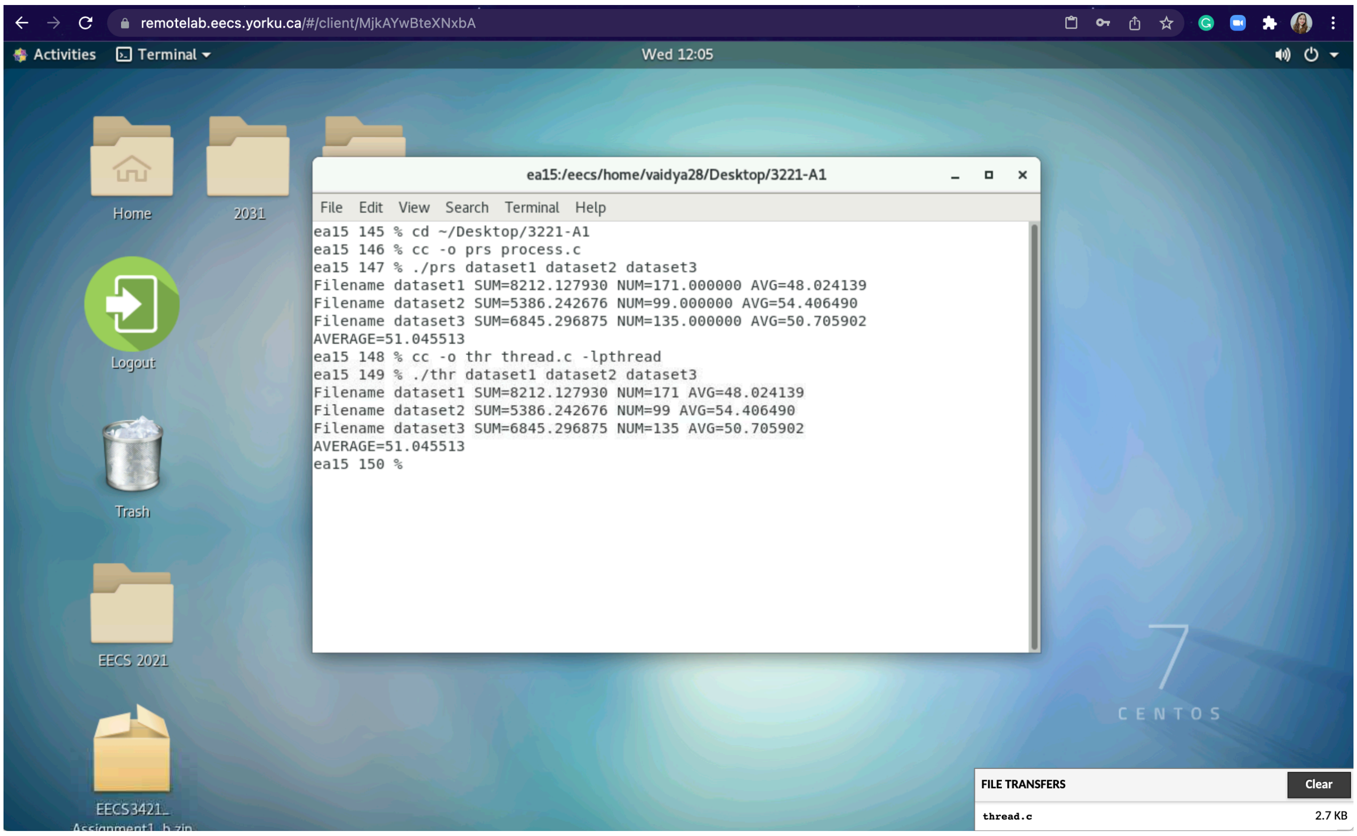
process.c	- System calls: fork, exit, pipe - Parent activates child process based on number of datasets - Child processes = number of datasets - Child reads files, calculates sum and maintains counter (num) - Child returns sum and num to parent through a unix pipe - Parent computes averages for datasets and prints it
thread.c Using pthreads	- Main thread activates worker theads based on number of datasets - Worker threads read files, calculates sum and maintains counter (num) - Worker thread stores sum and num on shared memory accessed by main thread - Main computes averages for datasets and prints it

3. Assigned work/components not done successfully

None, I completed all the assigned components except the bonus one. My program outputs the correct response and uses all the components (like systems calls - fork, exit, pipe, etc and threading/multithreading).

4. Any other comments

Following is an attachment of the output I got when I ran program on EECS server:-


remotelab.eecs.yorku.ca/#/client/MjkAYwBteXNxbA
Activities
Terminal
Wed 12:05
Home
2031
Logout
Trash
EECS 2021
EECS3421_Assignment1_b.zip
ea15:/eecs/home/vaidya28/Desktop/3221-A1
File Edit View Search Terminal Help
7 CENTOS
FILE TRANSFERS
Clear
thread.c
2.7 KB

```
ea15 145 % cd ~/Desktop/3221-A1
ea15 146 % cc -o prs process.c
ea15 147 % ./prs dataset1 dataset2 dataset3
Filename dataset1 SUM=8212.127930 NUM=171.000000 AVG=48.024139
Filename dataset2 SUM=5386.242676 NUM=99.000000 AVG=54.406490
Filename dataset3 SUM=6845.296875 NUM=135.000000 AVG=50.705902
AVERAGE=51.045513
ea15 148 % cc -o thr thread.c -lpthread
ea15 149 % ./thr dataset1 dataset2 dataset3
Filename dataset1 SUM=8212.127930 NUM=171 AVG=48.024139
Filename dataset2 SUM=5386.242676 NUM=99 AVG=54.406490
Filename dataset3 SUM=6845.296875 NUM=135 AVG=50.705902
AVERAGE=51.045513
ea15 150 %
```