

# Homework 1

CSCE 5640 Operating System Design, Fall 2024

Due 09/21/2024 at 11:59pm

All programs must compile and execute on the CSE machines! This is an individual assignment.

1. (15) Briefly describe the semantics of the Unix system calls listed below. Place your descriptions in a plain text file named **syscalls.txt**
  - a. **getpid()** ;
  - b. **getppid()** ;
  - c. **open()** ;
  - d. **read()** ;
  - e. **write()** ;
  - f. **close()** ;
  - g. **chmod()** ;
2. (20) Write a program that uses each of the above system calls at least once. Comment your code and submit it in a file named **syscalls.cpp**
3. (20) Consider the evolution of computers in general and the increase in processor speed in particular. Suppose we were to design a system with a processor that is clocked at 3GHz. Further, let us suppose that signals in our system can propagate at 80% of the speed of light. Determine the maximum distance between the CPU and Memory (or Cache) if we want to access data within one processor cycle. Provide your answer and a description of how you arrived at this answer in a plain text file named **length.txt**.
4. (45) Write a small program that copies data from a file A to a file B. For different file sizes 1KB – 128MB (doubling the file size in each step). So, you will create 18 different files with file sizes 1KB, 2KB, 4 KB, 8KB and so on. Record the time your program requires to complete the copy when using **read()** and **write()** system calls for each byte that is being copied. Generate a graph that depicts the program performance (you can create a graph in excel). In the graph, x-axis should be different number of file sizes and y-axis should be the total time (in seconds) to complete copying files. Briefly describe what you observe and investigate if you can improve the program's performance. For this program it is imperative that file A and file B are located on the local disk. For submission,
  - a) Provide your source code in a file named **copying.cpp**
    - Note: you must use system calls (read and write) otherwise you will not get the points.
    - Your program should take file names (source and destination) from the command line and do not hard code the file names.
    - Add error handling codes such as file not available, error opening files etc.
    - Add comments in your code.
  - b) Provide your graph in a file named **copying.jpg**
  - c) Provide your description in a file named **copying.txt**

Note that you may use **truncate** command in Linux to create dummy files with specific file sizes. Also, you can use **time** command in Linux to collect exact time taken by your program to complete execution.

**Submission:**

- We will be using an electronic homework submission on Canvas to make sure that all students submit their programming tasks on time. You will submit both the program source code file(s) to the **Homework 1 dropbox** on Canvas by the due date and time.
- Make sure you submit the following files for this homework (do not submit a zip file):
  - `syscalls.txt`
  - `syscalls.cpp`
  - `length.txt`
  - `copying.cpp`
  - `copying.jpg`
  - `copying.txt`
- Program submissions will be checked using a code plagiarism tool against other solutions, including those found on the Internet, so please ensure that all work submitted is your own. Any student determined to have cheated will receive an **'F'** in the course and will be reported for an academic integrity violation.
- Your programs will be tested on our Linux CSE machines. So, make sure you test your programs on the CSE machines fully before you submit on canvas.
- Until you are comfortable working on our Linux CSE machines, as a safety precaution, do not edit your program (using vim or nano) after you have submitted your program where you might accidentally re-save the program, causing the timestamp on your file to be later than the due date. If you want to look (or work on it) after submitting it, make a copy of your submission and work off that copy. Should there be any issues with your submission, this timestamp on your code on the CSE machines will be used to validate when the program is completed.