CSc 332 - Operating Systems

Task 4 - System Calls Summary

Max Points: 30 **Due**: October 29, 2020 11:59 PM

PART 1: Simple Command Interpreter

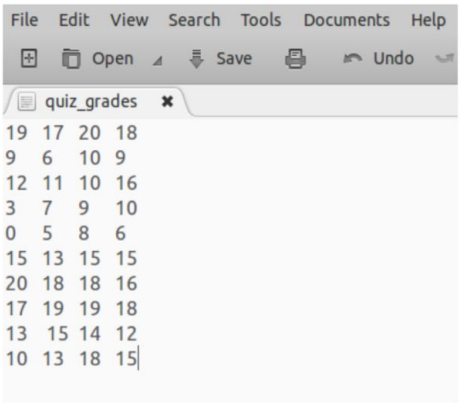
Recall: In Task 3, we worked with exec() system calls for specific commands such as date, and ls.

Write a special simple command interpreter that takes command and its arguments. This interpreter is a program where the main process creates a child process to execute the command using exec() family functions. After executing the command, it asks for a new command input (i.e., parent wait for child). The interpreter program will get terminated when the user enters quit.

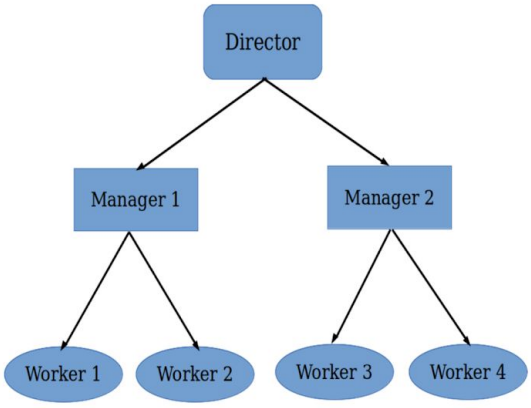
PART 2: Grade Calculator

There are 10 students enrolled in a course. The course covers x number of chapters from a textbook ($x > 1$). In each chapter y number of homework(s) are assigned ($y \geq 1$). The average grade for each homework in all the chapters need to be found out.

To solve this, write a program that has the main process as **Director** process, which reads a file containing grades of all homework of all chapters and creates **x** number of **Manager** processes. Each Manager process will take care of solving a chapter. Each manager process will create **y** number of **Worker** processes and pass one homework to each of them and they calculate and print the max and average. The input file should contain the data according to the value of x and y. For example, the input text file and the process tree for $x = 2$ and $y = 2$ will look like the following:



(a)



(b)

In the above fig a, we have 10 row for 10 students and each column has score for hws. First two columns has score of hw for chapter 1 while the last two has the score of hws for chapter 2.
The Director process is responsible for opening and closing the input text file. It stores the values in a two dimensional integer array with 10 rows. You may need to use the following C functions (in addition to the necessary file & process management system calls): fopen(), fscanf(), fseek(), fclose().

Submission Instructions

- Save your programs in a single folder. Make sure your programs compile and run without any errors.
- Email link to the code with subject line "Task 4 – CSc 332 – firstname lastname"

******