**ESOF 3655: Operating Systems**

**Project1**

Version 1.0, Last Edited 09/13/2024

Name of Students:

Date of Experiment:

# Part I: Objective

This project includes: Analytical questions regarding concepts discussed in class and practical exercises. The goal of the practical exercises in this project is to help students with hands-on experience of writing OS functions for processes management such as displaying running processes, logging running processes and killing processes using C programs.

# Part II: Equipment/Software

The practical exercises in this is project are designed for Windows; however, you can also use Mac or Linux, as long as you manage to find the equivalent commands.

The students are free to use any IDE/Compiler ( e.g. codeblocks (comes with the compiler), CodeLite, NetBeans, Eclipse etc.) that they are comfortable with. The programming language must be C.

# Part III: Guidelines

Due to the fact that students require some time to brush up on their C programming, this project will last for two weeks, the submission date is **September 27, 2024.** Students are encouraged to collaborate with each other, however, this is an individual project and each student is responsible for his/her submission.

1. Students are required to submit working code, tested in the lab or by other mean that allows the instructor to inspect the working code. Failure to do so will result in deducting marks or not getting any credit for the particular exercise.

2. The students may declare variable or function using any name they wish, however, naming conventions of good programming practice are highly recommended.

Some useful links:

<https://users.ece.cmu.edu/~eno/coding/CCodingStandard.html>

<http://homepages.inf.ed.ac.uk/dts/pm/Papers/nasa-c-style.pdf>

3. Students must provide comments for each line of their code clearly, showing the exact meaning of each line from start to finish (Failure to do so or providing inadequate comments will result in deducting marks)

4. Soft copy of the code is not required to be submitted, the students must copy and paste the code in their project answer document along with explanation of the output.

# Part IV: Getting Started with C

**This should be your start point for the upcoming project/lab tasks.**

You should dedicate a good amount of time to review the basics of C programming. Lab projects, to the most part, include C programming tasks. A beginner C programmer should do just fine with some extra time invested in reviewing the syntax and logic of programming.

There are ***plenty*** of resources to start C programming, examples can be found here:

- Google "C Programming Tutorial"

- Thousands of hits !!!!!!

<https://www.tutorialspoint.com/cprogramming/cprogramming_tutorial.pdf>

<http://phy.ntnu.edu.tw/~cchen/ctutor.pdf>

et....

YouTube is also recommended:

Type: "c programming tutorial" on YouTube's search engine, some good tutorials:

Sample of C programming tutorial series:

<https://www.youtube.com/playlist?list=PL6gx4Cwl9DGAKIXv8Yr6nhGJ9Vlcjyymq>

<https://www.youtube.com/watch?v=z8evnLrYm9I&list=PL0170B6E7DD6D8810&index=5>

<https://www.youtube.com/playlist?list=PLfVsf4Bjg79CZ5kHTiQHcm-l2q8j06ofd>

**Part IV: Description of Tasks**

**Task 1 (Concept Questions, Total Marks: 20)**

(a) (**5 Marks**) Describe three general methods for passing parameters to the operating system.

(b) (**5 Marks**) Describe the differences between symmetric and asymmetric multiprocessing. What are three advantages and one disadvantage of multiprocessor systems?

(c) (**5 Marks**) What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel

approach?

(d) (**5 Marks**) Select the correct answer**:**

(1) A clustered system \_\_\_\_.

A) gathers together multiple CPUs to accomplish computational work

B) is an operating system that provides file sharing across a network

C) is used when rigid time requirements are present

D) can only operate one application at a time

(2) What are some other terms for kernel mode?

A) supervisor mode

B) system mode

C) privileged mode

D) All of the above

(3) A \_\_\_\_\_ is an example of a systems program.

A) command interpreter

B) Web browser

C) text formatter

D) database system

(4) If a program terminates abnormally, a dump of memory may be examined by a \_\_\_\_ to determine the cause of the problem.

A) module

B) debugger

C) shell

D) control card

(5) A boot block \_\_\_\_.

A) typically only knows the location and length of the rest of the bootstrap program

B) typically is sophisticated enough to load the operating system and begin its execution

C) is composed of multiple disk blocks

D) is composed of multiple disk cylinders

**Task 2 (Programming, Total Marks: 30)**

Google and study the windows command prompt commands *tasklist* and *taskkill* and their various options.

**a)** **(2 marks)** Use *tasklist* to take a list of all your processes and save them into a file *log.txt*. At the very least, the following 4 attributes for each process must be logged: name, process ID (PID), memory usage, and CPU time. Attach this log as your answer

**b)** **(2 marks)** Open a notepad document. Then open a command prompt and obtain the PID of the notepad process. Use the *taskkill* command to close the notepad process. In your answer show the commands you used to achieve the goal and explain what happens for each command.

**c)** **(8 marks)** Write a C program that would launch a notepad, obtain its PID, wait for 3 seconds, and then kill the notepad process.

**d)** **(8 marks)** Write a C program that would constantly watch all processes (say once every second), and if it encountered a notepad, it would kill it immediately.

**e)** **(10 marks)** Write a C program that would constantly watch all processes (say once every second), and if it encountered any process that is using more than 200 Kbytes of memory or 3 minutes of CPU time, it would report them to the user, and ask the user if it wants to kill any of them. It would show something like this to the user:

Using more than 200K:

1- notepad

2- cs352.exe

Using more than 3 minutes:

3- word.exe

4- firefox

Enter a number to kill, or wait for the next refresh ->

Good Luck,