EE 463 | **Winter 2023**

Operating Systems

Homework #4

Date: 17/06/1444
10/01/2023

Dr. A. M. Al-Qasimi

Due: 17/01/2023

Objectives:

The objective of this assignment is to exercise the student's understanding of the concepts of processes, threads, and CPU scheduling which are central to how Operating Systems work.

What to Do:

1) **Read** Chapters 3, 4 and 5 of the textbook *Operating System Concepts*, 10th edition. These chapters introduce concepts that are in the heart of every modern operating system. Read over this, but don't try to memorize it! The following assignment questions will test your understanding of the important topics.

2) **Answer the following questions**:

Q_1. Describe the differences among short-term, medium-term, and long-term scheduling.

Q_2. Describe the actions taken by a kernel to context-switch between processes.

Q_3. Give an example of a situation in which ordinary pipes are more suitable than named pipes and an example of a situation in which named pipes are more suitable than ordinary pipes.

Q_4. What are the benefits and the disadvantages of each of the following? Consider both the system level and the programmer level.

a. Synchronous and asynchronous communication
b. Automatic and explicit buffering
c. Send by copy and send by reference
d. Fixed-sized and variable-sized messages

Q_5. Provide two programming examples in which multithreading does **not** provide better performance than a single-threaded Solution.

Q_6. Can a multithreaded solution using multiple user-level threads achieve better performance on a multiprocessor system than on a single processor system? Explain.

Q_7. A system with two dual-core processors has four processors available for scheduling. A CPU-intensive application is running on this system. All input is performed at program start-up, when a single file must be opened. Similarly, all output is performed just before the program terminates, when the program results must be written to a single file. Between startup and termination, the program is entirely CPU-bound. Your task is to improve the performance of this application by multithreading it. The application runs on a system that uses the one-to-one threading model (each user thread maps to a kernel thread).

a. How many threads will you create to perform the input and output? Explain.
b. How many threads will you create for the CPU-intensive portion of the application? Explain.

Q_8. Why is it important for the scheduler to distinguish I/O-bound programs from CPU-bound programs?

Q_9. One technique for implementing **lottery scheduling** works by assigning processes lottery tickets, which are used for allocating CPU time. Whenever a scheduling decision has to be made, a lottery ticket is chosen at random, and the process holding that ticket gets the CPU. The BTV operating system implements lottery scheduling by holding a lottery 50 times each second, with each lottery winner getting 20 milliseconds of CPU time (20 milliseconds × 50 = 1 second). Describe how the BTV scheduler can ensure that higher-priority threads receive more attention from the CPU than lower-priority threads.

Q_{10}. Consider a system running *ten* I/O-bound tasks and *one* CPU-bound task. Assume that the I/O-bound tasks issue an I/O operation once for every *one* millisecond of CPU computing and that each I/O operation takes *10* milliseconds to complete. Also assume that the context-switching overhead is *0.1* millisecond and that all processes are long-running tasks. Find the CPU utilization for a round-robin scheduler when:

a. The time quantum is 1 millisecond
b. The time quantum is 10 milliseconds