CECS 326 Assignment 2 (5 points)

Due: October 3, 2019 by class time on BeachBoard

When a user launches a program for execution, the OS creates a user process to execute it. Many OS, including Linux, provides the mechanism for a process to create child processes to run concurrently with the parent process which creates them. These concurrent processes may need to communicate between them. One way Linux supports interprocess communication is message queue. A message queue must first be acquired from the operating system by calling **msgget**. Control operations, e.g., remove, can be performed on an existing message queue by calling **msgctl**. Processes with appropriate permissions may send and/or receive messages via the message queue by calling **msgsnd** and **msgrcv**. Please consult the man pages of these system calls for details.

This assignment is aimed at helping you familiarize with the process management and message queue mechanism on Linux.

Do the following for this assignment:

1. Compile and run the *use\_msgQ.cpp* program shown in the Processes lecture notes.
2. Try to understand what the program does, and annotate the program with adequate amount of comments throughout the program to explain what the program does. Detailed explanations are required for all system calls used in the program for process creation/management as well as for message queue management and message passing. Explanations on the system calls must include meaning of each parameter used in the call.