**CECS 326 Sec01**

Operating Systems

Elias Woldie (ID 027805724)

Assignment 2

Due Date: 10/05/2023

Submission Date: 10/07/2023

**Program Description**

1. **What all the programs together are designed to do?**

The programs (**master.cpp**, **sender.cpp**, and **receiver.cpp**) collectively demonstrates inter-process communication (IPC) through message queues in a UNIX-like operating system. The main program (**master.cpp**) initiates the communication by creating specified numbers of sender and receiver child processes. Each sender is tasked with collecting a message from the user, sending it to the message queue, and then awaiting an acknowledgment from a receiver. Conversely, receivers pick up messages from the queue, process them, and send back an acknowledgment. The master process oversees the creation and management of these child processes and ensures orderly execution and cleanup.

1. **What each individual program does?**
   1. **master.cpp:**
      1. This is the controller program. It first sets up a message queue and then spawns a specified number of sender and receiver child processes.
      2. It waits for all child processes to finish their operations.
      3. Finally, it cleans up by removing the message queue and then terminates.
   2. **sender.cpp**
      1. Represents the sending entity in this communication setup.
      2. Upon starting, it retrieves the message queue ID and prompts the user for a message.
      3. After the user provides a message, it sends the message to the message queue.
      4. It then waits for and displays an acknowledgment from a receiver before terminating.
   3. **receiver.cpp**
      1. Represents the receiving entity in this communication setup.
      2. It starts by retrieving the message queue ID.
      3. It then waits for a message from the sender in the message queue.
      4. Upon receiving a message, it processes (displays) the message.
      5. It sends back an acknowledgment to the sender and then terminates.