**CECS 326-01**

Operating Systems

Yunis Nabiyev (ID 030973967)

Assignment 1

Due Date: 10/01/2024

Submission Date: 9/29/2024

Program Description

This assignment demonstrates how information can be passed between processes using SystemV commands (msgget(), msgsnd(), msgrcv(), msgctl(), etc.). The three programs show these concepts by creating children and passing information from sender to receiver (and back for acknowledgements) processes through the message queue system. The master program takes in the number of receivers to create, creates the message queue with msgget(), and creates as many receivers as needed and 1 sender in addition. The master outputs information about itself, its status as it executes, as well as its children. Once all the children are created, it waits for the sender process to exit (waitpid()). The sender process is created inside of master and is passed via arguments the message queue ID and how many receivers to expect. It then waits for the user to enter a message and enter which receiver to send it to. Once the message is sent using msgsnd(), it waits for a message on the appropriate channel (type) to come back from the receiver. The receiver(s) is also created inside of the master and passed the message queue ID and its receiver ID. It waits for messages to come into the System V message queue with their applicable channels (example: receiver 1 reads messages of type 1). Once a message is retrieved, the receiver sends an acknowledgment message back to the sender and waits for the next message. The sender sees this acknowledgement and continues. In my program I added the ability to send more messages after each one, and the ability to terminate the receivers only when messages are no longer going to be sent. Once the messaging is done, the master removes the message queue and terminates alongside its children. As stated before, this assignment shows how process can communicate with one another, and demonstrates the lifecycle of queues and their messages.