Mason Shreffler
Documentation for OS Assignment 1:

The table values are shared between the producer and consumer processes using shared memory. The producer writes values to the table, which can hold up to two values, while the consumer reads and consumes them from the shared memory. Only one process can access the table at a time, and this is controlled by semaphores to prevent concurrent access. The producer is prevented from writing to a full table, while the consumer is prevented from consuming from an empty table.
In case a process is unable to perform its action, it sleeps for a random period, while the other process performs its action. Both processes use a mutex to ensure mutual exclusion during their critical sections. After a process completes its critical section, it signals the other process to allow it to perform its action using semaphores.
The program continues to produce and consume items for a specified number of times, which is determined by the loop variable in both programs. It is essential that these variables have equivalent values to ensure that the program functions correctly.

To run the program, use the following instructions:

```
$ gcc producer.c -pthread -lrt -o producer

$ gcc consumer.c -pthread -lrt -o consumer

$ ./producer & ./consumer &
```