CSCI-7645: Practice Problem Set 6

Fairleigh Dickinson University Vancouver
Fall 2023

1. Create a POSIX shared memory consisting of an array of 10 items where each item is a `Person` with a `firstName` (20 bytes) and a `lastName` (20 bytes). Implement a writer process that writes 10 items to this shared memory and a reader process that reads them from the shared memory and displays them to screen. Ensure that race conditions are avoided.

2. Consider two processes: a producer and a consumer, and a shared variable `numMessages`. The shared variable starts with an initial value of 0. The producer increases the value of `numMessages` in steps of 1 in an infinite loop until it reaches a maximum value of 1024. When the producer tries to increment the value of `numMessages` beyond 1024, it must block until the value is less than 1024. The consumer decrements the value of `numMessages` in steps of 1 in an infinite loop until it reaches a minimum value of 0. When the consumer tries to decrement the value of `numMessages` below 0, it must block until the value becomes positive. When the producer and consumer are run simultaneously, the following conditions must be satisfied:

   (a) There must be no race conditions.

   (b) `numMessages` must not exceed 1024.

   (c) `numMessages` must not be negative.

   Implement the producer and consumer using POSIX shared memory and POSIX semaphores.