# Assignment

**A.** Write a program in which parent process creates a child process, child process takes in put to generate two matrices and put in the shared memory and parent process reads the matrices in the shared memory and then calculates the product matrix using multiple threads and stores in the shared memory, client process reads product matrix and prints on the console.

**B.** Write a program that creates three child processes and three pipes to communicate with each process. Each child process reads from different serial line and sends the characters read back to the parent process through a pipe. The parent process outputs all characters received. A child terminates when a '#' character is received. The parent terminates after all three children have terminated. (hint: send-pipe and receive-pipe primitives can be used)

**C.** Write a program that spawns *N* threads, where n is a program argument. These threads access a shared counter (initialized to 0) in a loop. In each iteration, they read the counter to a local variable, increment it, and store it back to the counter. When all threads complete 50000 iterations each, the program stops and prints the value of the shared counter. **PROTECT THE SHARED COUNTER**, so that no two threads modify the counter at the same time. Run the program on 2, 4, and 8, threads.

**D.** The reader writer problem: A number of readers may simultaneously be reading from a file. Only one writer at a time may write to file, and no reader can be reading while a writer is writing. Using semaphores, Write solution to the reader writers problem that gives priority to writers.

**E.** In Section 5.7.1 (page: 219), a semaphore-based solution to the producer– consumer problem using a bounded buffer is given. In this project, you will design a programming solution to the bounded-buffer problem using the producer and consumer processes shown in Figures 5.9 and 5.10.

**F.** Modify the socket-based date server (Figure 3.21) in Chapter 3 so that the server services each client request in a separate thread.