|  |  |
| --- | --- |
| 1 | Write a program to sum the elements of an array (list) using 4 threads. Let each thread add quarter of the array. Assume that the size of the array is a multiple of 4. Use global variable “total\_sum” which gets updated by each thread once it computes partial sum. Use lock when global\_variable is getting updated. |
| 2 | Write a program where one process acts like a server and the other the client. Each time the client communicates a number, the server returns the square of it. Avoid racing using semaphores. Avoid catering to the same request twice. Let parent process (client) read the number from keyboard and child process (server) computes the square and prints it and this must happen continuously until the number sent is -1. If the number sent is -1, terminate the child process and end the program. Make use of shared variable. |
| 3 | Implement two-way communication between parent and child processes using:   1. Pipe 2. Queue |

**PAP Week 10 Lab Assignments**