

Lab 9 - Message Passing Lab

Coding Exercises

1. Multi thread the *PrimeSortServer* from the previous lab (Lab 8) so that the server can handle many concurrent clients
2. Modify the *PrimeSortServer* to find and return the maximum prime number in the *ArrayList* which has been received from the client. Make sure the *PrimeSortServer* is still multithreaded
3. How can you test if the multithreaded *PrimeSortServer* is able to handle multiple concurrent client requests at a time?
4. Implement the Producer and Consumer using Synchronous Message Passing as covered in the Message Passing lecture, the Consumer will be the server and the producer will produce integers from 1-10 and sends them to server one after another. Use object serialization from send integers from the producer to the consumer.
5. Design and implement the Producer and Consumer using asynchronous message passing as covered in the last slide of the Message Passing lecture. Do NOT code unless you have a clear design i.e, a UML class diagram. Also discuss and consider the suitability of using RMI or sockets for implement this type of asynchronous message passing system. For example which one is more convenient for clients to call, which has has better abstraction and is easier to implement.