The Tutorial Questions and Lab Projects of Week 3

Tutorial Questions

1. The `Election` interface provides two remote methods:

   `vote`: with two parameters through which the client supplies the name of a candidate (a string) and the 'voter's number' (an integer used to ensure each user votes once only). The voter's numbers are allocated sparsely from the range of integers to make them hard to guess.

   `result`: with two parameters through which the server supplies the client with the name of a candidate and the number of votes for that candidate.

   Which of the parameters of these two procedures are input and which are output parameters?

2. Discuss the invocation semantics that can be achieved when the request-reply protocol is implemented over a TCP/IP connection, which guarantees that data is delivered in the order sent, without loss or duplication. Take into account all of the conditions causing a connection to be broken.

3. A request-reply protocol is implemented over a communication service with omission failures to provide at-least-once RMI invocation semantics. In the first case the implementor assumes an asynchronous distributed system. In the second case the implementor assumes that the maximum time for the communication and the execution of a remote method is T. In what way does the latter assumption simplify the implementation?

4. A client makes remote procedure calls to a server. The client takes 5 milliseconds to compute the arguments for each request, and the server takes 10 milliseconds to process each request. The local operating system processing time for each send or receive operation is 0.5 milliseconds, and the network time to transmit each request or reply message is 3 milliseconds. Marshalling or unmarshalling takes 0.5 milliseconds per message.

   Calculate the time taken by the client to generate and return from two requests:

   (i) if it is single-threaded, and

   (ii) if it has two threads that can make requests concurrently on a single processor.

   You can ignore context-switching times. Is there a need for asynchronous RPC if client and server processes are threaded?

5. Briefly discuss the advantages and disadvantages of the middleware approach.

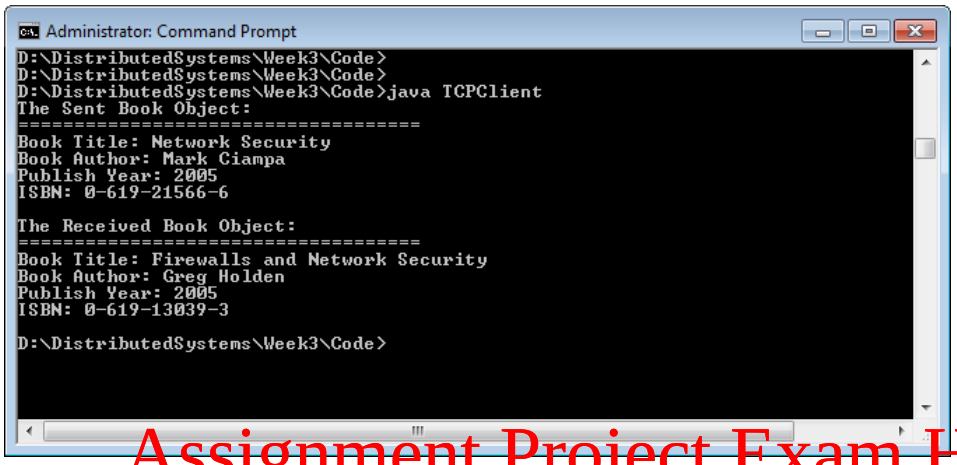
Lab Projects

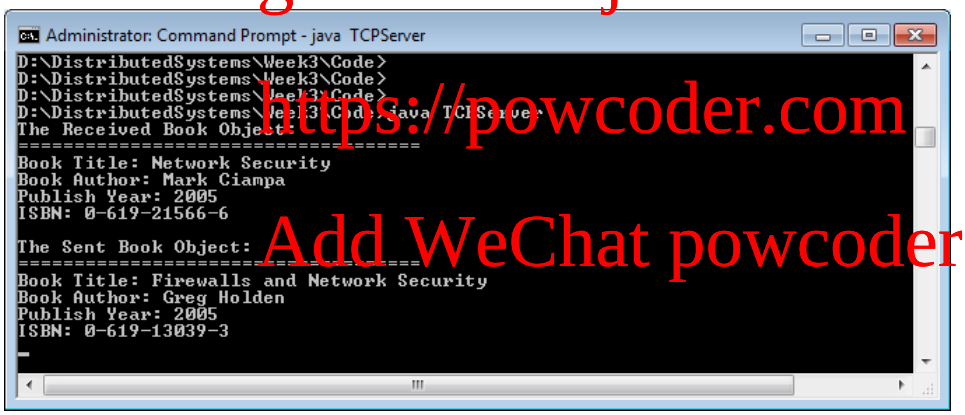
Java objects can be transmitted as parameters between a client program and a server program, which are running on two different JVMs (Java Virtual Machines). This feature somehow facilitates access transparency. In this project, you are required to use Java APIs to complete the following task.

1. Discuss how to transfer a serialized object between a client process and a server process via TCP protocol.

2. Use the TCP example of Week-2 lecture and the Book object of Week-2 lab tasks to develop a client/server application to demonstrate object transfer between remote processes.

   The following screenshots demonstrate that the TCPClient sends a book object to the TCPServer, which replies another book object to the former.





.

Reference:

1. Java object serialization specification from Sun Microsystems at:

   http://docs.oracle.com/javase/1.3/docs/guide/serialization/

2. Java APIs on Socket, ServerSocket, `ObjectOutputStream` and `ObjectInputStream`.

   http://docs.oracle.com/javase/6/docs/api/