CSS 434
Lab Work 3a: RMI

Professor: Munehiro Fukuda
Lab work date: See the syllabus

---

1. Purpose

This laboratory work intends to let you understand how to code and run RMI programs, so that you can focus on your HW3's code design.

2. Statement of Work

We will re-implement Lab1a: EchoServer.java with RMI. However, we are not using the original TCP-based EchoServer.java at all.

1. Look at ~css434/lab3a. You can access Client.java and Server.java. The client identifies the server through Naming.lookup() and calls its echo() function. On the other hand, the server instantiates a ServerImplementation object and registers it through Naming.rebind(). Whenever a client calls the server's echo(), the server returns the message back to the client's receiveMessage().
2. You can look at ServerInterface.java. Implement ServerImplementation.java that should include its constructor and echo() function. The echo() function should just call the client's receiveMessage().
3. Thereafter, look at ClientInterface.java and implement ClientImplementation.java. The latter should include its constructor and receiveMessage() fucntion. The receiveMessage() function should just print out a given message through System.out.println().

3. Related Materials

- To take a quick review for RMI, see the slides: p19-25 of RMI.ppt
- To take a snapshot of the entire X server screen, type:

```
import -window root X.jpeg
```

4. What to Turn in

Turn in the following materials to Canvas by the due date of Program 3:

1. Your ClientImplementation.java and ServerImplementation.java
2. Your execution output, (e.g., X.jpeg)