OOSD: Practice Problems 10

1. Write a complete Java program, called *PortTester*, that takes a hostname as its command line parameter, prompts the user for a port number to test, reads an integer from the keyboard, and tells the user whether there is a server running on that port on the given machine. Use only Java IO.

   Sample run (User typing is in boldface):

   % **java PortTester l-lnx101.divms,uiowa.edu**
   Which port do you want to test? **7**
   l-lnx101.divms.uiowa.edu has a server running on port 7

2. Many Unix machines have a server running on TCP port 13 that provides the time of day for that machine as a line of text. Write a *complete* Java program that acts as a client for one of these servers. The machine that is being connected to is named as a command line argument. Here is a sample execution:

   % java DayClient athena-dist.mit.edu
   Mon Dec 17 16:01:31 2002

3. Write a program that provides a Time Server on port 1313. When a client connects to the server, it returns a string containing the current data and time as given by the *toString* method for a newly created java.util.Date object. The server should respond to all clients until it is terminated using control-c.

4. Use telnet to connect to and use the following servers.

   Echo Server at cse.buffalo.edu
   Date Server at time-A.timefreq.bldrdoc.gov
   Finger Server at cse.psu.edu (enter an additional return)
   SMTP Server at mail.divms.uiowa.edu
   POP3 Server at blue.weeg.uiowa.edu

5. Write a *complete* Java program *DudeServer.java* that acts as a simple server on port 12345. Whenever a client connects to the server, the server returns the string "Hey, dude" and closes the connection. The server should be designed to run forever, but it will handle only one client at a time.

6. Write a program that sends email using SMTP. Get the server name, the addressee, and the message from the user and send the email using nobody@sun.com as the sender. Send this email to yourself only.

7. Add code to the Time37 program to calculate the number of day, hours, minutes, and seconds that have elapsed since January 1, 1900 GMT.

8. Change the Time37 program to use a DataInputStream to read the four bytes as an **int**. If the number is negative (it will be), use **long** arithmetic to make if positive by adding 2^{32}.