Assignment 3: The Socket Interface

CS349
Fall 2000

Allen B. Downey
Computer Science Deparment

Due: Tuesday 19 September

The purpose of this assignment is to play around with the Socket object in Java and compare it to the socket interface described in the book (which is for C in a UNIX environment).

0.1 Java Sockets

First, read Section 1.3 of the textbook. Then go to the following web page and read the tutorial about the Java socket object.

`http://java.sun.com/docs/books/tutorial/networking/sockets/`

1. Go to your favorite Java development environment and create programs named `KnockKnockClient` and `KnockKnockServer`. You can pick up the code from the links in the tutorial or from

   `http://rocky.wellesley.edu/cs349/code/hw03/`

   You will also have to pick up the file `KnockKnockProtocol.java`, although you might want to make that an internal class for `KnockKnockServer`.

2. Compile the two programs and run them according to the instructions in the tutorial. Don't forget that you will have to change the name of the host where the client goes looking for a server. This means you have to know the name of the machine where you are running the program. If you don't know, and you don't know how to find out, ask me. Enjoy the delightful knock-knock jokes!

3. You might also want to try running the multi-threaded version of the server. I have it running on rocky, so you can connect your client to the server there if you want to.

0.2 A proxy server

A proxy server is a server that acts as an intermediate between a client and a server. Instead of connecting directly to the server, the client connect to the proxy. The proxy forwards requests from the client to the server and then forwards the replies from the server back to the client.

In the real world, proxy servers are useful for privacy (the client needn't know who the server is and vice versa) and security (the proxy might sit outside a firewall and provide carefully controlled and monitored access to the things inside the firewall).

For this assignment, the proxy server will just be silly.

1. Using the code you have as a template, write a proxy server. You might want the proxy server to listen to a different port number, so that the proxy server and the real server can run on the same machine.

2. Test your program by starting the real server, then the proxy server, then the client. The behavior seen by the server and the client should be exactly the same; you should not have to make any changes to either (except changing the port number the client connects to).

SUGGESTION: Debugging distributed programs like this is hard. The more time you spend planning and desk-checking your program, the less time this assignment will take.

0.3 A little comparison shopping

The Java socket interface and the C/UNIX socket interface provide basically the same set of features, but the C/UNIX interface requires the programmer to do more things explicitly, whereas the Java interface hides many of these details.

Compare the code on pages 45-48 of the text with the KnockKnock code from the Java tutorial. Give three examples of things that have to be done explicitly in C/UNIX that do not appear in the Java code.

0.4 What to turn in

One you have your proxy server working, take a minute to clean up the code. Remove anything unnecessary, make sure your variables have meaningful names, and format the code to make it easy to read.

Add comments to the code, at an appropriate level of abstraction. For example, do not write comments that are completely redundant with the code:

```
        // add seven to a and store the result in x
int x = a + 7;
```

Instead, explain at a high level what each chunk of code does:

```
 // keep looping until server quits
         while (true) {
            // get a line from the server and see if we're done
    String fromServer = serverIn.readLine();
    if (fromServer == null) break;
    if (fromServer.equals("Bye.")) break;
            ...
```

Write a paragraph or two comparing the Java interface to the C/UNIX interface. Include these paragraphs with your code.

Please staple all pages together and write your name, and the fact that this is Homework 3, in the upper right hand corner of the page.