CECS 472 **Homework 10** **Fall 2013**

Purpose: To gain an in-depth understanding of a threaded concurrent connection-oriented server (Chapter 12).

Turn your browserd server into a `threaded` concurrent connection-oriented server.

The `tbrowserd` server will listen on your personal well-known TCP port. The server will accept as many clients as want to connect. Each time it accepts a connection from a client, it *generate a new thread* to handle that client. When the client closes the socket, the thread procedure should return.

Note: The switch from fork to thread is the only difference between the two assignments.

For you testing, use the client from the previous homework. Do not modify the client.

Submission: You must submit a printout of your server. In addition you must place the source code for the serve in your home directory in a file called `tbrowserd.c`.

This is to be a simple conversion; do not use semaphores, do not add statistics.

Note: Since this is a single process, when a `c` is done it will change directory for the process, which means that all clients will be moved into the new directory. Don't try to fix this behavior, the fix is very complex. What it means is for this particular type of server, a threaded implementation might not be the best choice. But I didn't want you to write a whole new server, so we will just tolerate this behavior.

Testing note: Make sure that you can connect with a second client, that is, that the exit of the first client doesnot shut down the server.

Nathan Pickrell 3 October 2013 (Week 6 Lecture 2) Due: 8 October 2013 (Week 7 Lecture 1)