**CSC 4200/CSC5200**

**Computer Networks, Spring 2015**

**Programming Assignment 3**

**Assign Date: April 8, 2015, Due: April 20, at class**

For this assignment you will write a network version of the tic-tac-toe game. If you don’t know how to play the game, then ask your friends, siblings, parents or the instructor. The objectives of this assignment are: 1) Gain experience in developing an application layer protocol. 2) Gain experience in developing socket based networked application.

Two players will run your program on two different machines to play with each other. Your program should be able to run both as client and server. When executed the program will try to run as client 1st, and if unsuccessful it will run as server. Your program should take the name of the remote machine as command line argument and try to connect to a predefined port number (assuming the other player is already running his program, which is listening on the predefined port) on the remote machine. If unsuccessful the program should start listening on the predefined port for the other program to connect. When two programs are able to connect the client should send some message for handshaking and the server should acknowledge. Then the game should proceed. Once a player makes a move the game board on the remote machine should be updated and the player should see the move. Once a game is over (tie or win by either player) the players should be able to exchange message to decide whether to play the game again or not. Graphics based game board display and mouse based user input is not required (text based display and keyboard based input is OK).

I have intentionally kept the rules (how to decide who will make the 1st move, how the other party will know about it etc.) of the game vague. You will design and develop the application level protocol for the game. You may use C/C++, java, or python to develop your program. You have to develop the program for UNIX environment. Describe the protocol. To describe the protocol you have to define and describe the syntax and semantics of the messages of the protocol. Also you have to define and describe actions of processes (server/client) in response to each message. You should have messages for handshaking, play termination etc. Your protocol should be unambiguous, complete, and clearly defined. The idea is anybody should be able read the protocol and develop a program in any language of their choice and that program should be able to play with the program developed by you.

**What you will turn in**

For this assignment, you will turn in zipped source files on ilearn. Additionally, submit a README file containing detailed instructions on how to compile and run you program. You will also submit a report (both printed copy and electronic version) describing the protocol (Syntax and semantics of the messages, actions to be taken by the server and the client when they receive a message).

**Grading:**

Protocol Description: 30 Points

Clearly written README file: 10 Points

Program: 70 Points

Total: 100 Points