Project 3 – A Networked Boggle Protocol Implementation

Due date of implementation – October 28th (at 11:59:59pm)

Students may work on this project alone, in a group of two, or in a group of three. Students who work alone will later implement a game client and server in the same programming language. Students in groups of two must implement a game client and server in different programming languages. Students in groups of three must implement one server and two clients, where the two clients are in different programming languages.

Deliverables:

- **Boggle client code:** This should be only the code (not executable).
- **Boggle server code:** This should be only the code (not executable).
- **Readme.txt document:** This should contain 1) names of your group members, 2) how you split up the work, 3) any sources used to help you complete your assignment, 4) any challenges you encountered, 5) example working gameplay (copy and paste the output of the terminals).

Intro

Boggle © is a challenging word-finding game that can be played in as little as 3 minutes and can be played by two or more players. You can find the official rules of the game at this website: http://www.hasbro.com/common/instruct/boggle.pdf . (Ignore the Boggle challenge cube.)

Our goal is to turn Boggle into a multi-player networked game consisting of a single server program which will interact with two or more clients. New players can start up their clients and join a game whenever it is in the Stage 1 "joining mode", but they cannot join a game already in progress.

This project assignment will be broken up into two phases. Phase 1 is to design the client/server protocol on paper. Phase 2 is to actually implement the client and server programs. This specification discusses Phase 2.

Client/Server Implementation

Start by modifying the starter code for the client and server programs in the language of your choice according to the number of people in your group. Here are some additional things you may want to know to implement the game:

Real Boggle Dice

The real Boggle game comes with sixteen letter cubes. Each cube has six sides with a letter on each side. The letters on each cube are not random because the English language is not random. Instead, they were chosen in such a way that common letters come up more often and it is easier to get a good mix of vowels and consonants.

The following lists all of the letters on all six faces of each of the sixteen cubes. You should decide on an appropriate way to represent this information in your program and declare it accordingly.

AAEEGN, ABBJOO, ACHOPS, AFFKPS
AOOTTW, CIMOTU, DEILRX, DELRVY
DISTTY, EEGHNW, EEINSU, EHRTVW
EIOSST, ELRTTY, HIMNQU, HLNNRZ

Timing

Some phases of the game rely on timers. You will want to look up how to implement this in your language of choice. One way to go about this is to take advantage of the fact that each server has a main loop that is constantly looping. You can use your programming language's equivalent of the gettimeofday() call to check if it is time to move from game stage 1 to stage 2, or from stage 2 to stage 3.

Select

You will not have to change any of the parameters to select from the sample code. Only use the sockets in the select read lists.

Unexpected quitting

Any of the clients will quit with a CTRL+C. When this happens, the servers detect the disconnection and remove that particular client socket. If this happens in the middle of game play, the game should still progress with the remaining players. If no players are left, the server should quit.

Turning in the assignment

You can submit the assignment on eLearning. Go to this class and navigate to the project 3 submission area. Please submit your files one at a time in the required places.