CSCI 4220 Assignment 1

Zeroconf Guessing Game

Due Date: Wednesday, March 1, 11:59:59 PM

We've all played the game, "Guess the Number." For this assignment, we will be building a "Guess the Number" server we can `telnet` into. This will be single-client only (i.e., an iterative server as we discussed in lecture). Additionally, it should support Zeroconf/Bonjour.

Upon connecting, a new number between 1 and 100 should be randomly generated. At this point, the client (you) should be able to enter a new guess using the command `GUESS <n>` where `n` is your numeric guess. At this point, one of three things *should* happen: you receive the response, `GREATER`, `SMALLER`, or `CORRECT`. Of course, properly written servers must also be able to handle unexpected situations.

Assuming you receive `GREATER`, you should increase your next guess and repeat the `GUESS <n>` command, likewise for `SMALLER`. Upon guessing the correct value, you should receive the response, `CORRECT` along with additional output stating, `GREAT GUESSING` if your guess count was less than $log_2(100) - 1$, `BETTER LUCK NEXT TIME` if your guess count was greater than $log_2(100) + 1$, or `AVERAGE` otherwise. At this point, the server should terminate its connection.

How will you find the port information for your "Guess the Number" game? It should use Zeroconf to register itself. Subsequently, you should be able to find the information about your service by resolving it. Make sure you can see this information appear on your Zeroconf browser. Please only use the `"local"` domain. During development the type of your server should be `"_gtn._tcp"`. For the name, you should use your RCS/CS username.

Unlike most iterative servers, this one will require the use of the `select()` system call due to the requirement of Zeroconf/Bonjour. The sample code from the Zeroconf book and 6.21/6.22 of our textbook may be helpful.

Invalid guesses should not count towards the total number of guesses. Output of invalid syntax is demonstrated below. Additionally, the server should not exit upon finishing game.

Example output below. User input is in bold.

```
laprej@loki$ dns-sd -L "Justin's Game" _gtn._tcp local
Lookup Justin's Game._gtn._tcp.local
DATE: ---Sun 19 Feb 2017---
13:18:18.738  ...STARTING...
13:18:18.739   Justin's\032Game._gtn._tcp.local.  can  be  reached  at  loki.local.:53942
(interface 4)
^C
laprej@loki$ telnet localhost 53942
Trying ::1...
telnet: connect to address ::1: Connection refused
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
```

```
guess 50
SMALLER
guess 25
SMALLER
guess 12
GREATER
guest 20
???
guess 20
SMALLER
guess 15
CORRECT
GREAT GUESSING
Connection closed by foreign host.
```