

CS270 - System Software

Lab Assignment #7

Spring 2021

The purpose of this assignment is to give you some experience using the sockets API.

For this exercise, you are to create a tournament of the game of "Rock-Scissors-Paper." Your game should consist of three programs - two "client" programs and a "server" program. The two clients represent the two players, while the server is the "referee" who scores the games. One client should be written to accept a choice from you, while the second client will randomly generate a choice; in other words, you will be playing against the computer. Once each client makes a choice, they will transmit their choices to the server via socket connections, who will announce the winner and keep score.

The client program that represents you should prompt you for a guess, which can be one of:

- r - rock
- s - scissors
- p - paper
- q - quit the game

The second client, representing the computer, will also make a guess. Both clients will then send their guesses to the server. The server will also keep a count of the number of time each player won.

To begin the tournament, you should start the server and the two clients, at which time the two sockets needed for the game should be established.

One IP address - 127.0.0.1 - is designated to be the "local loopback" address - this address can be used for a machine to communicate with itself. Using this address causes any communication to go through the entire sockets process. This address facilitates testing, since both clients and server can reside on the same machine. Alternately, a client on a machine can access a server on the same machine by specifying its own IP address - in this case, the request "leaves" the machine and then "reenters" the same machine, which is usually a slightly more rigorous test than local loopback.

Recall that sockets use ports to determine which process a client wants to connect with. On most machines, for security purposes most port requests are refused - the port is closed. On our machine, ports 4500-4525 are open and can be used.

An example dialogue is shown below - those lines marked with a C: are generated by the computer client, a Y: are generated by the client you are interacting with, and R: by the referee server program. The \$ is the system prompt:

```
$ referee & # the server program
$ computer & # the computer client
$ you # client representing you
Y: Enter your guess (r, s, p) or q to quit: r
C: The computer makes its guess
R: You chose rock, computer chose paper, computer wins!

Y: Enter your guess (r, s, p) or q to quit: s
C: The computer makes its guess
R: You chose scissors, computer chose scissors, a draw!

Y: Enter your guess (r, s, p) or q to quit: p
C: The computer makes its guess
R: You chose paper, computer chose rock, you win!

Y: Enter your guess (r, s, p) or q to quit: r
C: The computer makes its guess
R: You chose rock, computer chose scissors, you win!

Y: Enter your guess (r, s, p) or q to quit: q
R: Final results: you = 2, computer = 1, draw = 1

R: thanks for playing!
```

