**Course**: Principals of Software Development – ENSF 409

**Lab 6**

**Instructor**: M. Moshirpour

**Student Name**: Mitchell Sawatzky

**Date Submitted**: Feb 26, 2016

Exercise A

Server.java

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import java.io.PrintWriter;

import java.net.ServerSocket;

import java.net.Socket;

/\*\*

\* Plaindrome validation server class

\* @author Mitchell Sawatzky

\* @version 1.0

\* @since Feb, 2016

\*/

public class Server {

private PrintWriter socketOut;

private BufferedReader socketIn;

private ServerSocket palinServerSocket;

private Socket palinSocket;

/\*\*

\* Constructs a server object

\* @constructor

\* @param portNumber - the port to listen on

\*/

public Server (int portNumber) {

try {

palinServerSocket = new ServerSocket(portNumber);

System.out.println("Server running on port " + portNumber + "...");

palinSocket = palinServerSocket.accept();

socketIn = new BufferedReader(new InputStreamReader(palinSocket.getInputStream()));

socketOut = new PrintWriter(palinSocket.getOutputStream(), true);

} catch (IOException e) {

System.err.println("Could not start server.");

System.err.println(e.getStackTrace());

System.exit(1);

}

}

public static void main (String[] args) {

Server serv = new Server(9898);

serv.communicate();

}

/\*\*

\* Server listener function

\*/

public void communicate () {

String line = "";

while (!line.equals("QUIT")) {

try {

line = socketIn.readLine();

if (validatePalindrome(line)) {

socketOut.println(line + " is a palindrome...");

} else {

socketOut.println(line + " is NOT a palindrome...");

}

} catch (IOException e) {

System.err.println("Socket write error: " + e);

}

}

try {

socketIn.close();

socketOut.close();

palinServerSocket.close();

} catch (IOException e) {

System.err.println("Error closing sockets: " + e);

}

}

/\*\*

\* Determines if a string is a palindrome or not

\* @param str - the string to test

\* @returns true if str is a palindrome, false otherwise.

\*/

public boolean validatePalindrome (String str) {

String rev = "";

for (int i = str.length() - 1; i > -1; i--) {

rev += str.charAt(i);

}

if (rev.equals(str)) {

return true;

} else {

return false;

}

}

}

Terminal output for Client:

Mitchell@ttys003 22:31 {1} [6]$ java Client

please enter a word:

radar

radar

radar is a palindrome...

please enter a word:

121

121

121 is a palindrome...

please enter a word:

red

red

red is NOT a palindrome...

please enter a word:

QUIT

QUIT

QUIT is NOT a palindrome...

Terminal output for Server:

Mitchell@ttys002 22:31 {0} [6]$ java Server

Server running on port 9898...