

## CECS 277 – Lecture 22 – Threaded Networked Connections

A networked connection can use a thread so that can continuously send and receive data. Below is an example of a instant messenger program that uses a thread to watch for incoming data from the client, and then another thread to watch for incoming data from the server.

**Example:** A Thread that receives input from a Server/Client and echoes it to the console.

```
import java.net.*;
import java.io.*;
class NetThread implements Runnable{
    Thread t;
    Socket sock;
    String source;
    NetThread(Socket s, String sc){
        sock=s;
        t=new Thread(this);
        t.start();
        source =sc;
    }
    public void run(){
        try{
            BufferedReader in =new BufferedReader(new
            InputStreamReader(sock.getInputStream()));
            while(true){
                String line=in.readLine();
                System.out.println(source+": "+line);
            }
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}
```

### Server: Messenger Server

```
import java.net.*;
import java.io.*;
import java.util.Scanner;
public class Serverchat{
    public static void main(String args[]){
        try{
            ServerSocket server=new ServerSocket(1235);
            System.out.println("Waiting...");
            Socket sock=server.accept();
            NetThread s=new NetThread(sock, "client");
            System.out.println("Connected: ");
            Scanner in=new Scanner(System.in);
```

```

        PrintStream o=new PrintStream(sock.getOutputStream());
        while(true){
            String line =in.nextLine();
            o.println(line);
        }
    }catch(Exception e){
        System.out.println(e);
    }
}
}

```

### **Client: Messenger Client**

```

import java.net.*;
import java.io.*;
import java.util.Scanner;
public class Clientsock {
    public static void main(String args[]){
        try{
            System.out.println("Waiting...");
            Socket sock=new Socket("localhost",1235);
            System.out.println("Connected: ");
            NetThread c=new NetThread(sock, "server");
            Scanner in=new Scanner(System.in);
            PrintStream o=new PrintStream(sock.getOutputStream());
            while(true){
                String line =in.nextLine();
                o.println(line);
            }
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}

/* Output of Server
Waiting...
Connected:
client: Hello
Hi
*/

/* Output of Client
Waiting...
Connected:
Hello
server: Hi
*/

```