Programming Assignment 1: UDP Ping Client/Server program

(modified form the book)

Ping is a network utility tool that is used to test reachability to a particular host on a TCP/IP network. It works by sending a packet to a specified host and waiting for a reply. Ping can be used for troubleshooting to test connectivity and determine response time between two IP host machines. The standard Ping program uses ICMP (Internet Control Message Protocol).

However, in this project you will create a pair of Ping Client/Server program built on UDP. Your ping program is to send **10 ping messages** to the target server over UDP. For each message your client is to determine and print RTT when the corresponding ping message is returned. Since UDP is an unreliable protocol a packet sent by the client or server may be lost. For this reason, the client cannot wait indefinitely for a reply to a ping message. You should have the client wait up to **one second** for a reply from the server; if no reply is received, the client should assume that the packet was lost and print a message accordingly.

Server code: The Ping server code is given below.

```
/**
 * PingServer.java
 */

class PingServer {
   public static void main(String args[]) throws Exception
   {
      DatagramSocket serverSocket = new DatagramSocket(2014); //server will run on port #2014
      byte[]  receiveData = new byte[1024];
    //Processing loop
      while(true)
        {
          //create a random number generator for use in packet loss simulation
            Random  random =  new Random();
          //create a datagram packet  to hold incoming UDP packet
           DatagramPacket receivePacket = new DatagramPacket(receiveData,  receiveData.length);
         // get the client message
           serverSocket.receive(receivePacket);
           String sentence = new String(receivePacket.getData(),0, receivePacket.getLength());
           InetAddress IPAddress = receivePacket.getAddress(); //get client's IP
           int port = receivePacket.getPort(); //get client's port #
          //print out the clients's IP address, port number and the message
           System.out.println("client's port #  = " + port);
           System.out.println("client'sIP  = " +IPAddress);
           System.out.println("client's message  = " +sentence);
           //capitalize the message from the client
            String capitalizedSentence = sentence.toUpperCase();
            //simulate the packet loss
            int rand =random.nextInt(10);//a random number in the range of 0 to10
           // if rand is less than 4 we consider the packet lost and do not reply
           if  (rand < 4) {
                       System .out.println("Reply not sent");
                        continue;
                       }
           //otherwise, the server responds
            byte[] sendData = capitalizedSentence.getBytes();
            DatagramPacket sendPacket =new DatagramPacket(sendData, sendData.length, IPAddress,
port);
            serverSocket.send(sendPacket);
           System.out.println("Reply to the client sent");
          }//while
     }//main
}
```

You should study the server code carefully so that you can write the corresponding ping client program. Your ping client program should sends 10 ping requests to the server. After sending each request client waits up to one second for reply from the server. If no reply is received during this time period, client times out. In this case you should print out the time out message.

Specifically, your client program should do the following:

(1)Print out the current time and the ping attempt number. Send the ping message using UDP.
(2) Print the response message from server, if any (the server will capitalize the message sent by the client and send it back to the client). Print out the elapsed time in microseconds.
(3) Calculate and print the round trip time (RTT), in microseconds, of each packet, if server responses.
(4) Otherwise, print “Request timed out”.