# Solutions to Assignment 2 Questions

1. **Problem P1, Chapter 2**
    a. False
    b. True
    c. False
    d. False
2. **Problem P5, Chapter 2**
    a.
        1. The document request was `http://gaia.cs.umass.edu/cs453/index.html`. The "`Host:`"-field indicates the server's name and `/cs453/index.html` indicates the file name.
        2. The browser is running HTTP version 1.1, as indicated just before the first `<cr><lf>`-pair
    b.
        1. The browser is requesting a persistent connection, as indicated by the `Connection: keep-alive`.
        2. This is a trick question. This information is not contained in an HTTP message anywhere. So there is no way to tell this from looking at the exchange of HTTP messages alone. One would need information from the IP datagrams (that carried the TCP segment that carried the HTTP GET request) to answer this question.
3. **Problem P4, Chapter 2**
    a. The status code of 200 and the phrase OK indicate that the server was able to locate the document successfully. The reply was provided on Tuesday, 07 Mar 2006 12:39:45 GMT (Greenwich Mean Time).
    b. The document index.html was last modified on Saturday 10 Dec 2005 18:27:46 GMT
    c. There are 3874 bytes in the document being returned
    d. The first five bytes of the returned document are : `<!doc`. The server agreed to a persistent connection, as indicated by the `Connection: Keep-Alive` field.
4. **Problem P8, Chapter 2**
    a. The total amount of time to get the IP address is $RTT_1 + RTT_2 + \cdots + RTT_n$. Once the IP address is known, $RTT_0$ elapses to set up the TCP connection and another $RTT_0$ elapses to request and receive the small object. The total response time is $2RTT_0 + RTT_1 + RTT_2 + \cdots + RTT_n$.

## 5. Problem P14, Chapter 2

**TCPServer.java**

```
import java.io.*;
import java.net.*;

class TCPServer {
     public static void main(String args[]) throws Exception {
          String clientSentence;

          ServerSocket welcomeSocket
          = new ServerSocket(6789);

          while(true) {
               Socket connectionSocket
               = welcomeSocket.accept();

               BufferedReader inFromClient
               = new BufferedReader(
                    new inputStreamReader(
                         connectionSocket.getInputStream()));

               clientSentence = inFromClient.readLine();

               System.out.println("RECEIVED FROM CLIENT: "
               + clientSentence + "\n");
          }
     }
}
```

## 6. Problem P15, Chapter 2

a. **C:** dele 1
   **C:** retr 2
   **S:** (blah blah ...
   **S:** ..... blah)
   **C:** dele 2
   **C:** quit
   **S:** +OK POP3 server signing of

b. **C:** retr 2
   **S:** blah blah ...
   **S:** ..... blah
   **S:** .
   **C:** quit
   **C:** +OK POP3 server signing of

c. **C:** `list`
**S:** `1 498`
**S:** `2 912`
**S:** `.`
**C:** `retr 1`
**S:** `blah .....`
**S:** `.....blah`
**S:** `.`
**C:** `retr 2`
**S:** `blah blah ...`
**S:** `..........blah`
**S:** `.`
**C:** `quit`
**S:** `+OK POP3 server signing off`