Problem 1 (R1). List five nonproprietary Internet applications and the application- layer protocols that they use.

Application	Application-layer protocols
Electronic mail	SMTP [RFC 5321]
Remote terminal access	Telnet [RFC 854]
Web	HTTP [RFC 2616]
File transfer	FTP [RFC 959]
Streaming multimedia	HTTP

Problem 2 (R4). For a P2P file-sharing application, do you agree with the statement, "There is no notion of client and server sides of a communication session"? Why or why not?

I don't agree with the statement because in the book it says "In a P2P file-sharing system, a file is transferred from a process in one peer to a process in another peer. For each pair of communicating processes, we typically label one of the two processes as the client and the other process as the server." which against the statement.

Problem 3 (R5). What information is used by a process running on one host to identify a process running on another host?

To identify a process, two pieces of information need to be specified:

(1) the address of the host and (IP address)

(2) an identifier that specifies the receiving process in the destination host. (Port number)

Problem 4 (R7). Referring to Figure 2.4 in the textbook, we see that none of the applications listed in Figure 2.4 requires both no data loss and timing. Can you conceive of an application that requires no data loss and that is also highly time- sensitive?

I think application that using finical data such as Bloomberg terminal require Both no data loss and Time sensitive since it require precise for the data and real time transfer.

Problem 5 (R9). Recall that TCP can be enhanced with SSL to provide process-to- process security services, including encryption. Does SSL operate at the transport layer or the application layer? If the application developer wants TCP to be enhanced with SSL, what does the developer have to do?

The SSL is an enhancement of TCP with the enhancements being implemented in the Application layer.

When a developer wants TCP to be enhanced with SSL he need to include SSL code in both the client and server sides of the application.

Problem 6 (R15). Why is it said that FTP sends control information "out-of-band"?

FTP uses two parallel TCP connections to transfer a file, a control connection and a data connection. Because FTP uses a separate control connection, FTP is said to send its control information **out-of-band**

Problem 7 (R20). Look over your received emails, and examine the header of a message sent from a user with an .edu email address. Is it possible to determine from the header the IP address of the host from which the message was sent? Do the same for a message sent from a Gmail account.

Yes it is possible to determine the IP address from emails sent from .edu. And could not do so with emails sent from gmail.

Problem 8 (R21). In BitTorrent, suppose Alice provides chunks to Bob throughout a 30-second interval. Will Bob necessarily return the favor and provide chunks to Alice in this same interval? Why or why not?

No, Even Alice is sending data to Bob she is not necessary become one of Bob's top four uploaders.

Problem 9 (R23). What is an overlay network? Does it include routers? What are the edges in the overlay network?

In an overlay network, the peers form an abstract logical network which resides above the "underlay" computer network consisting of physical links, routers, and hosts. The links(edges) in an overlay network are not physical links, but are simply virtual liaisons between pairs of peers. There is no router included.

Problem 10 (R27). For the client-server application over TCP described in Section 2.7, why must the server program be executed before the client program? For the client- server application over UDP, why may the client program be executed before the server program?

Because when the client create its TCP socket it has to specific the address of the welcoming socket in the server. So if the server is not executed it will go wrong. However for the UDP it is not necessary to make the connection in the very beginning

Problem 11 (P1). True or false?

a. A user requests a Web page that consists of some text and three images. For this page, the client will send one request message and receive four response messages.

   False

b. Two distinct Web pages (for example, www.mit.edu/research.html and www.mit.edu/students.html) can be sent over the same persistent connection.
   True

c. With non-persistent connections between browser and origin server, it is possible for a single TCP segment to carry two distinct HTTP request messages.
False

d. The **Date:** header in the HTTP response message indicates when the object in the response was last modified.
False

e. HTTP response messages never have an empty message body.
False

Problem 12 (P2). Read RFC 959 for FTP. List all of the client commands that are supported by the RFC.

USER, PASS, CWD, CDUP, SMNT, REINITIALIZE, LOGOUT

Problem 13 (P6). Obtain the HTTP/1.1 specification (RFC2616). Answer the following questions:

a. Explain the mechanism used for signaling between the client and server to indicate that a persistent connection is being closed. Can the client, the server, or both signal the close of a connection?

This signaling takes place using the Connection header field Once a close has been signaled, the client MUST NOT send any more requests on that connection.
Both client and there server can close the connection

b. What encryption services are provided by HTTP?

HTTP does not provided by HTTP

c. Can a client open three or more simultaneous connections with a given server?

   Clients that use persistent connections SHOULD limit the number of simultaneous connections that they maintain to a given server. A single-user client SHOULD NOT maintain more than 2 connections with any server or proxy.

d. Either a server or a client may close a transport connection between them if either one detects the connection has been idle for some time. Is it possible that one side starts closing a connection while the other side is transmitting data via this connection? Explain.
   Yes It is possible From the server's point of view, the connection is being closed while it was idle, but from the client's point of view, a request is in progress.

Problem 14 (P14). How does SMTP mark the end of a message body? How about HTTP? Can HTTP use the same method as SMTP to mark the end of a message body? Explain.

SMTP mark end with a line only contains period. However HTTP use header to indicate length on the message. They cannot use same way because SMTP use only 7-bit ASCII but HTTP data do not have this restriction.

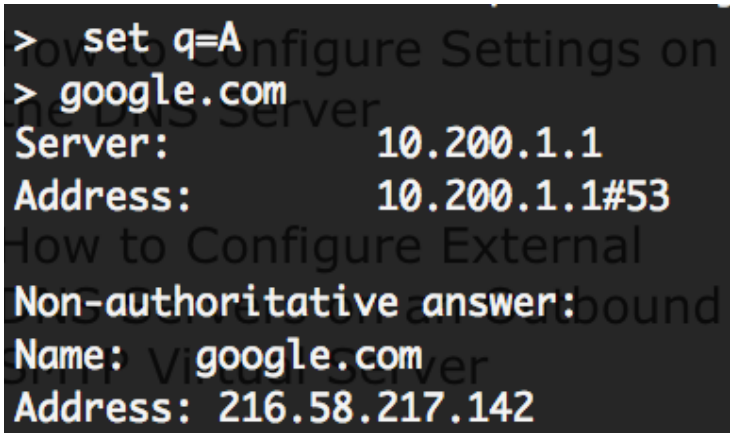
Problem 15 (P18). Answer the following questions:

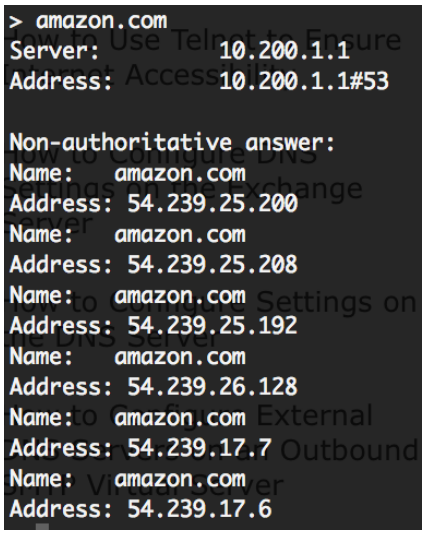
a. What is a *whois* database?

The Whois database is an online repository of information associated with registered domain names. It stores and publicly displays domain name information, such creation and expiration dates, the registrar of record, and its various contacts (registrant, billing, administrative, and technical).

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.
I use ICANN WHOIS database and ARIN to obtain DNS servers. Two DNS servers I got are www.google.com and www.amazon.com

c. Use nslookup on your localhost to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b). Try querying for Type A, NS, and MX reports. Summarize your findings.





```
> google.com
Server:         10.200.1.1
Address:        10.200.1.1#53

Non-authoritative answer:
google.com      mail exchanger = 10 aspmx.l.google.com.
google.com      mail exchanger = 40 alt3.aspmx.l.google.com.
google.com      mail exchanger = 50 alt4.aspmx.l.google.com.
google.com      mail exchanger = 20 alt1.aspmx.l.google.com.
google.com      mail exchanger = 30 alt2.aspmx.l.google.com.
```

```
> amazon.com
Server:         10.200.1.1
Address:        10.200.1.1#53

Non-authoritative answer:
amazon.com      mail exchanger = 5 amazon-smtp.amazon.com.
```

```
> set q=NS
> google.com
Server:         10.200.1.1
Address:        10.200.1.1#53

Non-authoritative answer:
google.com      nameserver = ns4.google.com.
google.com      nameserver = ns1.google.com.
google.com      nameserver = ns3.google.com.
google.com      nameserver = ns2.google.com.
```

```
> amazon.com
Server:         10.200.1.1
Address:        10.200.1.1#53

Non-authoritative answer:
amazon.com      nameserver = ns3.p31.dynect.net.
amazon.com      nameserver = ns1.p31.dynect.net.
amazon.com      nameserver = ns4.p31.dynect.net.
amazon.com      nameserver = pdns1.ultradns.net.
amazon.com      nameserver = ns2.p31.dynect.net.
amazon.com      nameserver = pdns6.ultradns.co.uk.
```

From the result I could conclude big website such as google and amazon provide multiple name server and mail exchanger.

d. Use nslookup to find a Webserver that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?

   www.amazon.com has multiple IP addresses and my school do have multiple IP

e. Use the ARIN whois database to determine the IP address range used by your university.
   there are several range for my university including:

   192.12.13.0 - 192.12.13.255,192.12.14.0 - 192.12.14.255, 128.220.0.0 - 128.220.255.255

f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.
   By knowing the IP address for a domain The attacker can easier do the “reflect attack” by using the IP address found in whois databases

g. Discuss why who is databases should be publicly available.

   It should be public because when register a new domain we need it as a reference.

Problem 16 (P21). Suppose that your department has a local DNS server for all computers in the department. You are an ordinary user (i.e., not a network/system administrator). Can you determine if an external Web site was likely accessed from a computer in your department a couple of seconds ago? Explain.

Yes we can use dig tool to help us solve the problem. By using dig we could get the query time for querying DNS nameservers. If some one in the department

Problem 17 (P25). Consider an overlay network with *N* active peers, with each pair of peers having an active TCP connection. Additionally, suppose that the TCP connections pass through a total of *M* routers. How many nodes and edges are there in the corresponding overlay network?

There are N node and N(N-1)/2 edges

Problem 18 (P26). Suppose Bob joins a BitTorrent torrent, but he does not want to upload any data to any other peers (so called free-riding).

a. Bob claims that he can receive a complete copy of the file that is shared by the swarm. Is Bob's claim possible? Why or why not?

   Yes it is possible since he can always get data through optimistic unchoking

b. Bob further claims that he can further make his "free-riding" more efficient by using a collection of multiple computers (with distinct IP addresses) in the computer lab in his department. How can he do that?
   Since the new comer for torrent do not have any resource so he/she can only download without upload anything. In this way Bob can always pretend he is a new comer and in this way do the free riding.

Problem 19 (P27). In the circular DHT example in Section 2.6.2, suppose that peer 3 learns that peer 5 has left. How does peer 3

update its successor state information? Which peer is now its first successor? Its second successor?

Peer 3's first successor will still be Peer4. When peer 5 left peer3 will ask it's first successor's first successor(peer 8) and mark peer3's second successor is peer8

Problem 20 (P34). We have seen that Internet TCP sockets treat the data being sent as a byte stream but UDP sockets recognize message boundaries. What are one advantage and one disadvantage of byte-oriented API versus having the API explicitly recognize and preserve application-defined message boundaries?

Advantage: for some application level protocol such as http do not have notion of message boundaries so using byte steam is easier for those protocol

Disadvantage: Need a unique way to separate two message. in other word make the end of a message.