Here are the sections of the textbook that the exam will cover.

```
Section 1.1
Section 1.3.1
Section 1.5
Section 1.7
Section 2.1
Sections 2.2.1 - 2.2.4
Section 2.7 (using Java)
```

Also, be familiar with the material from the class presentations. Read all the Readmes and be familiar with the code from the two zip files

```
Inter-Process Communication.zip
Network Communication.zip
```

1. Explain the difference between the terms internet and intranet?

2. What are the layers of the Internet protocol stack? What are the principle responsibilities of each of these layers (in just one or two sentences for each layer)?

3. Here is an IP address in binary form. 1011 0011 0001 0011 1010 1000 1110 1010 Write the IP address in dotted-decimal form.

4. What do IP addresses identify? What do port numbers identify?

5. UDP sits "on top of" IP. Describe one important feature of the UDP protocol that is not a part of the IP protocol. What problem does that new feature solve?

6. TCP also sits "on top of" IP. Describe two important features of the TCP protocol that are not part of the IP protocol. What problem does each of those new features solve?

7. It is not very accurate to say that Computer1 is communicating with Computer2 using TCP. Why? Make this statement more precise in two different ways.

8. Why might you send an IP packet to a computer without giving the packet any port number? (Who might the packet be for?)

9. What makes a computer into a "server"?

10. Suppose we have to design a client/server system where the server's responsibility is to receive and store data from clients. Under what circumstances would UDP be a better transport protocol that TCP? Under what circumstance would TCP be better than UDP? Explain you answers.

11. A computer at IP address 205.215.21.12 creates a ServerSocket on port 434 and calls accept() on the socket. A computer at IP address 205.215.43.34 creates a Datagram-Socket and uses it to send a DatagramPacket addressed to 205.215.21.12:434. What do you think will happen and why?

12. Identify as many parts of the following URL as you can.

    `http://computer1.bigcompany.edu:4242/where/was/that/file.scm?x=2+2&bob=a4s56d7v83n`

13. What would be the HTTP request message (request line and the most important request headers) for the following URL?

    `http://math.purduecal.edu:80/~rlkraft/cs40400/class.html#Oct01`

14. How does an HTTP server know when it has received the last request header?

15. How does an HTTP client know when it has received the last response header? How does the client know when it has received all of the entity body?

16. Suppose we have three processes, P1, P2, and P3 that communicate using TCP. The processes are supposed to form a ring in which P1 can send messages to P2, P2 can send messages to P3, and P3 can send messages to P1 (and the processes do not communicate in any other way). More specifically, suppose that when process P1 starts, it prompts a user at a console to type in a message, and then P1 sends that message to P2. P2 receives the message and prints it on its console and then prompts a user at the console to type a new message that P2 sends to P3. P3 receives its message, prints it to its console, prompts a user for a new message, and sends that to P1. P1 receives the message from P3, prints it to its console window, and prompts its user for a message to send to P2, etc. Write a Java pseudo-code outline of the steps that P1 must go through to implement this "ring protocol". How many sockets, and what kind of sockets, will P1 need? What information would P1 need at its startup time? Would the pseudo-code for processes P2 and P3 be the same or different than P1's?

17. Explain the relationship between a "program" and a "process". If you have a Java source file, say MyGreatIdea.java, how do you turn it into a program? How do you turn it into a process?

18. What is "inter-process communication" and why is it so important? What feature of modern operating system design makes inter-process communication difficult?

19. Describe the five kinds of inter-process communication we have discussed in class. How do they compare to each other (in terms of abilities and limitations)?

20. Why do modern operating systems give all processes a "standard input," a "standard output," and a "standard error"? Answer this question in terms the software design concept of "separation of concerns" (or "separation of responsibilities"). How does the existence of stdin, stdout, and stderr help to separate responsibilities?

21. When a process reads from its stdin, it is sometimes reading from a keyboard, other times it might be reading from a file, and other times it might be reading from a network connection. If the idea of stdin (and I/O redirection) was not available, how might a single program implement these three distinct abilities?

22. Write a single Windows cmd.exe command line that successively applies filter programs Filter1.exe and Filter2.exe to the data in file File1.dat and writes the results to the file File2.dat.

23. Write a single Windows cmd.exe command line that successively applies filter programs Filter1.class and Filter2.class to the data in file File1.dat and writes the final results to standard output and redirects all the error outputs (from both Filter1 and Filter2) to the file Errors.log.

24. When a process starts up and initializes itself, where should it look for initialization data? In what order should the various locations for initialization data be examined? What is the reason for choosing that particular order?

25. Write five lines of Java that creates a server socket, waits for a connection on that socket, reads one line from the connecting process, writes that line back to the process, and then closes the connection (but not the server socket).

26. Write five lines of Java code that creates a udp socket, waits for a message to arrive in the socket, prints the contents of the message to standard out, and returns the message to the process that sent it.