

1.
 - a. DNS Basics
 - i. Both
 - ii. 53
 - iii. 3
 - iv. Yes
 - v. Local DNS server
 - vi. Authoritative DNS server
 - vii. dns.enterprise.com
 - viii. 2
 - ix. dns.enterprise.com, 146.54.96.10
 - x. CNAME
 - xi. MX
 - xii. enterprise.com, mail.enterprise.com
 - xiii. Yes
 - b. DNS-Iterative vs Recursive
 - i. Iterative
 1. DNS Root
 2. DNS TLD
 3. DNS Authoritative
 4. A
 5. Iterative
 - ii. Recursive
 1. DNS Root
 2. DNS TLD
 3. DNS TLD
 4. A
 5. Iterative
 - c. HTTP Get
 - i. quotation6.htm
 - ii. HTTP/1.1
 - iii. True
 - iv. False
 - v. American English
 - vi. English
 - vii. False
 - viii. True
 - d. HTTP Response
 - i. HTTP/1.0
 - ii. Yes

- iii. 204
- iv. nonpersistent
- v. image/html
- vi. Apache/2.2.3
- vii. Yes
- e. Electronic Mail and SMTP
 - i. SMTP
 - ii. SMTP
 - iii. HTTP
 - iv. TCP
 - v. push
 - vi. pull
 - vii. 25
 - viii. 80
- f. A comparison of client-server and P2P file distribution delays
 - i. 480
 - ii. s
 - iii. 320
 - iv. c

2.

- a. R8: The four broad classes of services provided by a transfer protocol are reliable data transfer, throughput management/flow control, timeliness guarantees, and security. TCP provides reliable data transfer, and UDP is more suited to timeliness guarantees. However, neither contribute to throughput management or security.
- b. R11: HTTP, SMTP, and IMAP run on top of TCP rather than UDP as these are all protocols that must ensure data is transferred correctly without any issues. TCP guarantees reliable data transfer while UDP does not.
- c. P3: DNS and TCP are needed to fetch the unknown IP address. DNS also runs on UDP so that could be included as well.
- d. P7: $RTT_1 + RTT_2 + \dots + RTT_{\square} + 2RTT_0$
- e. P8:
 - i. $RTT_1 + RTT_2 + \dots + RTT_{\square} + 18RTT_0$
 - ii. $RTT_1 + RTT_2 + \dots + RTT_{\square} + 6RTT_0$
 - iii. $RTT_1 + RTT_2 + \dots + RTT_{\square} + 10RTT_0$

3. A Conditional HTTP GET request enables the client to check whether there is a cached copy of the desired resource. There should be an initial request, which is not conditional, that contains metadata specifically when the file was last modified. From there subsequent GET requests are conditional, as they check if the file has been modified since the initial metadata was returned.

4. Clarku.edu would register its name at a DNS server by firstly requesting a domain under .edu. Following that they would provide contact information, specifically the desired domain name,

authoritative server name, and IP address. If all looks good the authority over the .edu servers will create DNS records that map incoming traffic to Clark's authoritative name servers.

5. In the worst case 6 DNS servers will be queried. Starting from pc24, the local server cs.keio.ac.jp is first queried. From here we get to the root server ".". From there we go to the top level .edu server. From there we go to the yale.edu server and start making our way down this hierarchy. From there we go to cs.yale.edu followed by ai.cs.yale.edu which is the authoritative server for robot.ai.cs.yale.edu.