

1. a. False, since each connection transports exactly one request message and one response message, there are four connections, i.e., four request messages and four response messages.

1. b. False, since A single TCP segment can only carry one HTTP request messages.

1. c. False, the "Date" is the time at which the request was created and not when the object was last modified.

1. d. False, since some HTTP response messages have an empty message body.

2. The total amount of time to get the IP address is $RTT_1 + RTT_2 + \dots + RTT_n$.

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

$$2RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$$

3. a. $\overbrace{RTT_1 + RTT_2 + \dots + RTT_n}^{\text{IP address}} + \overbrace{2RTT_0}^{\text{TCP connection}} + \overbrace{8 \times 2RTT_0}^{8 \text{ small objects}}$
 $= 18RTT_0 + RTT_1 + RTT_2 + \dots + RTT_n$

3. b.

$$RTT_1 + \dots + RTT + 2RTT_0 + 2 \cdot 2RTT_0$$

$$= 6RTT_0 + RTT_1 + \dots + RTT_n$$

3. c.

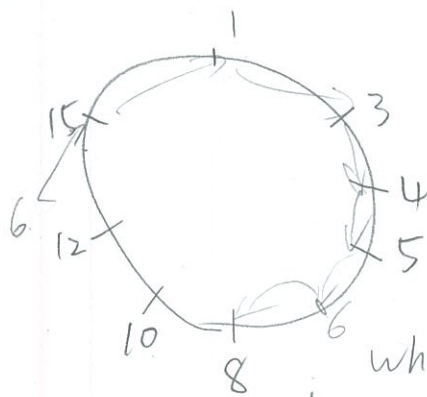
$$RTT_1 + \dots + R_n TT + 2RTT_0 + RTT_0$$

$$= 3RTT_0 + RTT_1 + \dots + R_n TT$$

4. a. Yes, His first claim is possible, as long as there are enough peers staying in the swarm for a long enough time. Bob can always receive data through optimistic unchoking by other peers.

4. b. His second claim is also true. He can run a client on each machine, and let each client do "free-riding", and combine those collected chunks from different machines into a single file. He can even write a small scheduling program to let different machines only asking for different chunks of the file. This is actually a kind of Sybil attack in P2P networks.

5.



first send peer 15 a message, saying "what will be peer 6's predecessor and successor?" This message gets forwarded through the DHT until it reaches peer 5, who realizes that it will be 6's predecessor and that its current successor, peer 8, will become 6's

successor. Next, peer 5 sends this predecessor and successor information back to 6. Peer 6 can now join the DHT by making peer 8 its successor and by notifying peer 5 that it should change its successor to 6