Computer Networks Final (108/1)

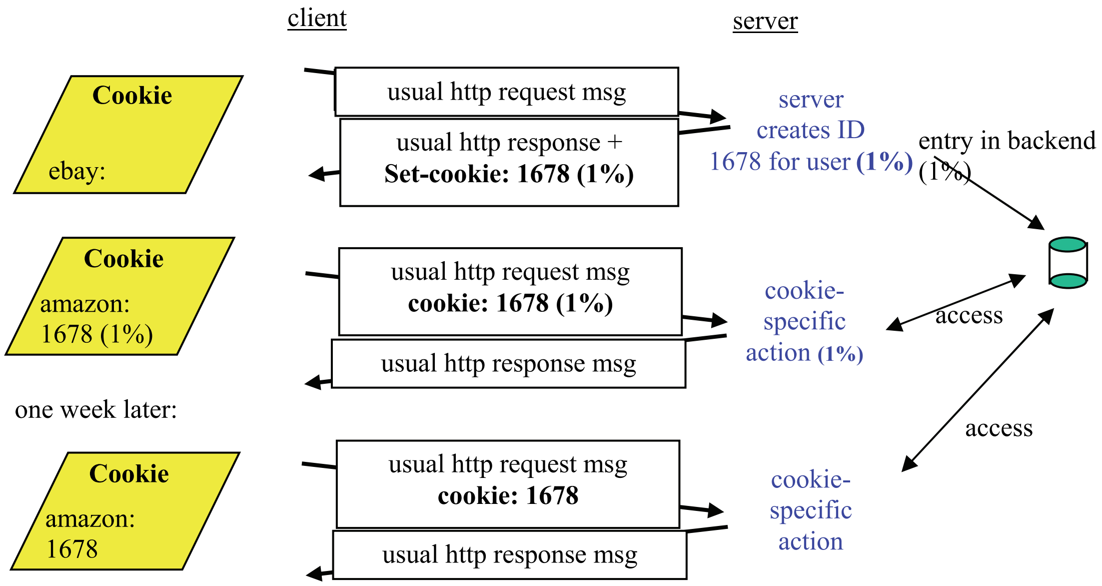
只寫答案而沒有解釋說明，扣一半分數

1. Describe detailed operations of HTTP cookie, web caching and conditional GET. (6*3=18%)（說明其用處，並畫圖加解釋每步驟）

Ans:

- ❑ cookie:

  when a user visits a specific web site for first time and initial HTTP requests arrives at site, site creates a unique ID and creates an entry in backend database for recording user states of this ID.
  => keep client's states (cookie-specific action)!



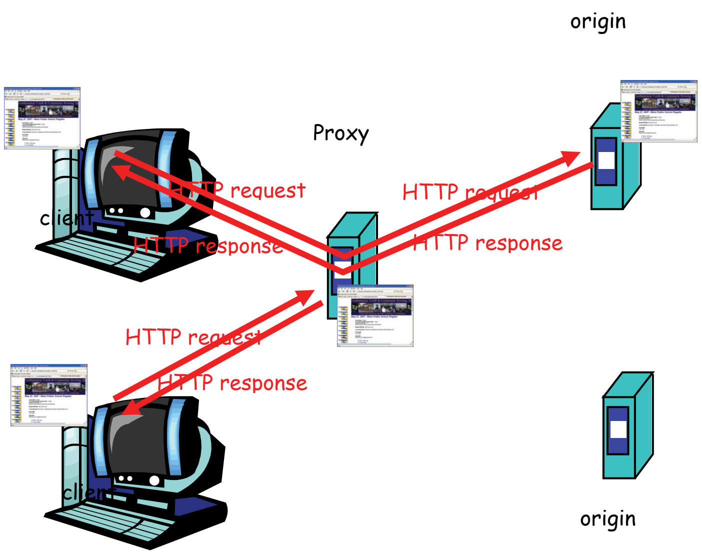
- ❑ web caching:
  - user sets browser: Web accesses via cache
  - browser sends all HTTP requests to cache (2%)

    if object in cache

    cache returns object (2%)

    else

    cache requests object from origin server, then returns object to client (2%)



conditional GET (6%)

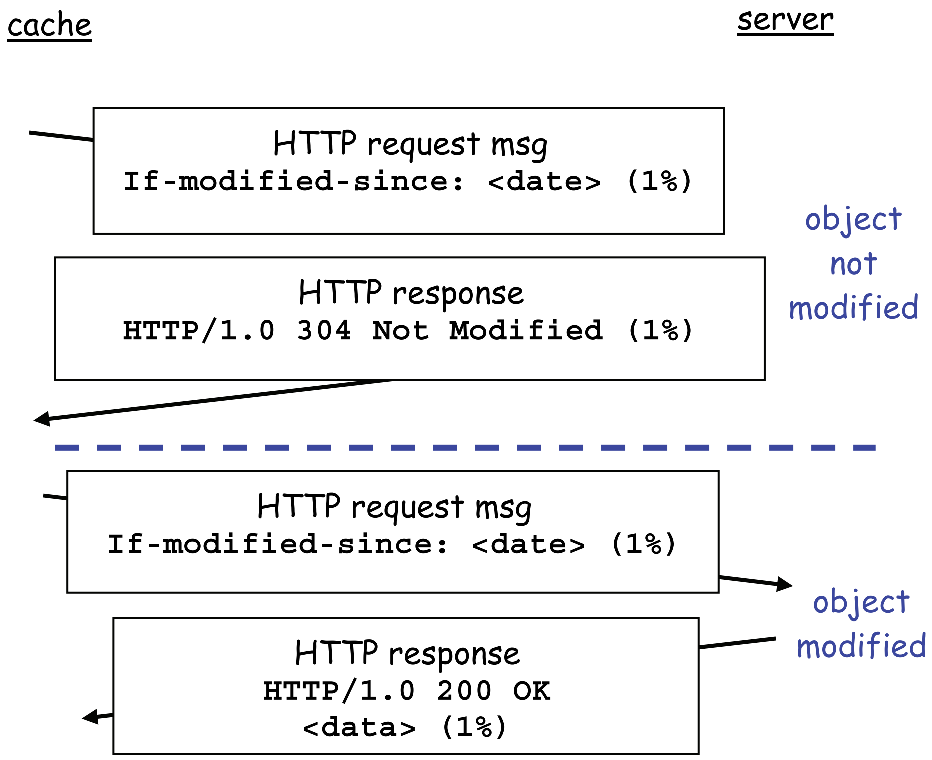
- Conditional GET: don't send object if cache has up-to-date cached version (1%) => reduce traffic loads (delays) on network links! (1%)

cache: specify date of cached copy in HTTP request (1%)

If-modified-since: <date> (1%)

server: response contains no object if cached copy is up-to-date: (1%)

HTTP/1.0 304 Not Modified (1%)



2. (a) Explain how TCP Fast Retransmit works. (3%) (18% total)
   (b) How TCP does its flow control? (3%)
   (c) What values are used by TCP to identify their sockets? (4%)
   (d) Describe four operations to provide reliable data transfer over channels with errors and loss? (8%)

Ans:

(a) Explain how TCP Fast Retransmit works. (3%)

If sender receives 3 ACKs for the same data, (1%) it supposes that segment after ACKed data was lost (1%): resend segment before timer expires (1%) (3% total)

(b) How TCP does its flow control? (3%)

Rcvr advertises spare room by including value of **RcvWindow** in segments (1%)

Sender limits unACKed data to **RcvWindow** (1%) for guaranteeing receive buffer doesn't overflow (1%)

(c) TCP socket identified by 4-tuple: (4%)

source IP address
source port number
dest IP address
dest port number

(d) sender adds *sequence number* to each pkt to detect duplicate pkts (2%)

receiver uses checksum to detect bit errors (2%)

receiver sends ACK with seq # of last pkt received OK (2%)

sender waits "reasonable" amount of time for ACK, retransmits if no ACK received in this time (2%)

3. Draw and write the flow of the TCP three way handshake to explain its operations. Suppose the initial sequence numbers of the client and the server are 90 and 50, respectively. 必須寫出三步驟的過程，在

Computer Networks Final (108/1)

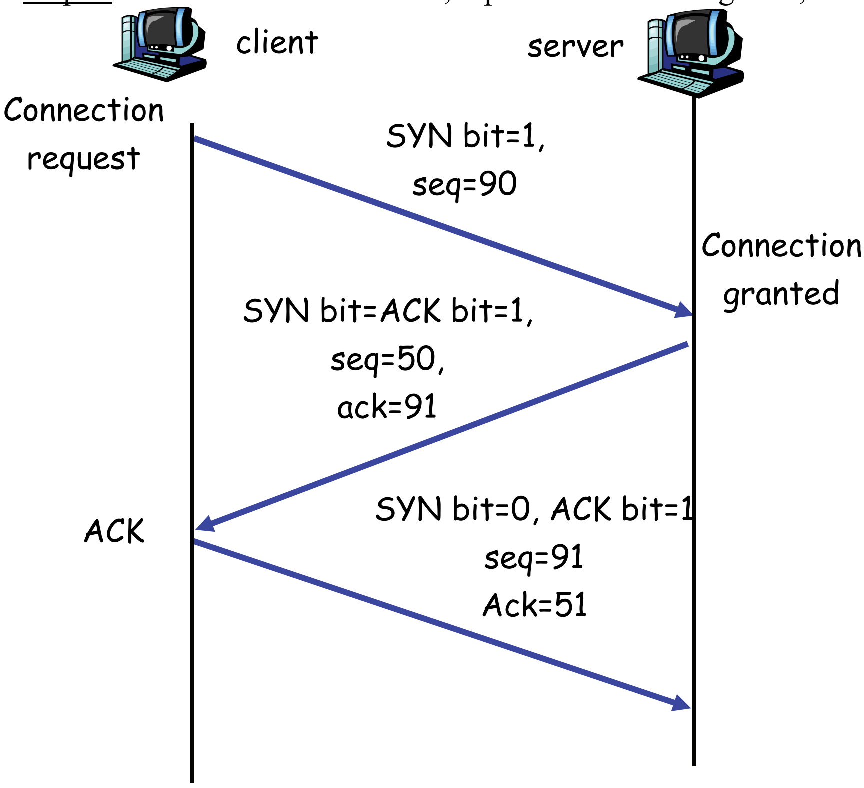
圖上分別清楚標示出 TCP 必要的 flag, sequence number, and ACK number. (10%)

Ans: Three way handshake:

Step 1: client host sends TCP SYN segment to server (搭配圖要正確 2%)

Step 2: server host receives SYN, replies with SYNACK segment (4%)

Step 3: client receives SYNACK, replies with ACK segment, which may contain data (4%)



上圖每個符號含內容 1 分，標示不全者，視狀況扣分，共 10 分

4. List and compare two pipelined transport protocols. (9%)

Ans:

Go-back-N (5%)

(1) ACK-only: always send ACK for correctly-received pkt with highest *in-order* seq # (1%)

(2) out-of-order pkt:

- discard (don't buffer) -> no receiver buffering! (1%)
- Re-ACK pkt with highest in-order seq # (1%)

(3) timeout(n): retransmit pkt n and all higher seq # pkts in window (1%)

(4) deliver in-order segments to upper layer. (1%)

Selective Repeat (4%)

(1) receiver *individually* acknowledges all correctly received pkts (1%)

(2) buffers out-of order pkts (1%)

(3) sender only resends pkts for which ACK not received when timeout (1%)

(4) deliver total in-order pkts to upper layer (1%)

5. Describe how TCP Reno does its congestion control. (8%)

Ans: (4%)

When **CongWin** is below **Threshold**, sender in slow-start phase, window grows exponentially (2%).

When **CongWin** is above **Threshold**, sender is in congestion-avoidance phase, window grows linearly (2%).

When a triple duplicate ACK occurs, **Threshold** set to **CongWin/2** and **CongWin** set to **Threshold** (2%).

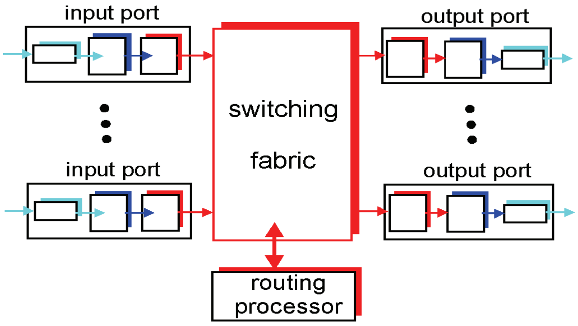
When timeout occurs, **Threshold** set to **CongWin/2** and **CongWin** is set to 1 MSS (2%).

6. (a) Draw a figure to show four components of a router (4%) (b) Draw and write three types of

switching fabrics with their names. (2% each) (10% total)

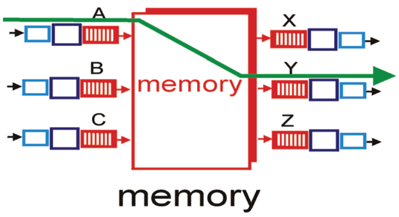
Ans:

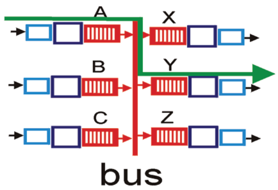
(a) (1% each, 4% total)

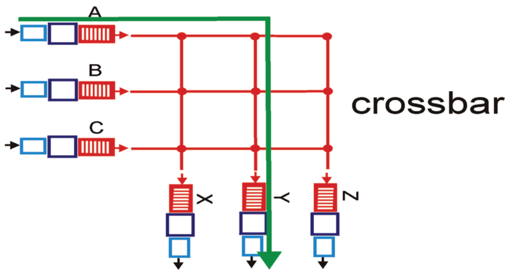


(b) (6%)

switching via memory; (6%)
switching via a bus; (6%)
switching via an interconnection network (6%)







7. Consider the TCP procedure for estimating RTT ($EstimatedRTT^{n} = \alpha \times SampleRTT^{n-1} + (1-\alpha) \times EstimatedRTT^{n-1}$).

(a) Why TCP uses this function? (2%)

(b) Let *SampleRTT*^{n} be the most recent sample RTT, let *SampleRTT*$^{n-1}$ be the next most recent sample RTT, and so on. Express *EstimatedRTT*^{n} in terms of n SampleRTTs if $EstimatedRTT^{1}=0$. (要有兩次疊代過程，直到 *SampleRTT*$^{n-2}$，每次各 2%)後寫出通式(以 summation 總和符號表示)(4%) (10% total)

Ans: (a) Exponential weighted moving average => influence of past sample decreases exponentially fast. 據測量出來的 SampleRTT，估計下一次的 EstimatedRTT，用來設定下一次的 Timeout 時間 (2%)

(b)

Computer Networks Final (108/1)

$\text{EstimatedRTT}^{n} = \alpha \times \text{SampleRTT}^{n-1} + (1-\alpha) \times \text{EstimatedRTT}^{n-1}$

$= \alpha \times \text{SampleRTT}^{n-1} + (1-\alpha) \times [\alpha \times \text{SampleRTT}^{n-2} + (1-\alpha) \times \text{EstimatedRTT}^{n-2}]$

$= \alpha \times \text{SampleRTT}^{n-1} + \alpha(1-\alpha) \times \text{SampleRTT}^{n-2} + (1-\alpha)^2 \times \text{EstimatedRTT}^{n-2}$

$= \alpha \times \text{SampleRTT}^{n-1} + \alpha(1-\alpha) \times \text{SampleRTT}^{n-2} + (1-\alpha)^2 \times [\alpha \times \text{SampleRTT}^{n-3} + (1-\alpha) \times \text{EstimatedRTT}^{n-3}]$

$= \alpha \times \text{SampleRTT}^{n-1} + \alpha(1-\alpha) \times \text{SampleRTT}^{n-2} + \alpha(1-\alpha)^2 \times \text{SampleRTT}^{n-3} + (1-\alpha)^4$
$\times \text{EstimatedRTT}^{n-3}]$

=..........

$= \alpha \times \text{SampleRTT}^{n-1} + \alpha(1-\alpha) \times \text{SampleRTT}^{n-2} + \alpha(1-\alpha)^2 \times \text{SampleRTT}^{n-3}$
$+ \cdots + \alpha(1-\alpha)^{n-2} \times \text{SampleRTT}^{n-(n-1)} + (1-\alpha)^{n-1} \times \text{EstimatedRTT}^{n-(n-1)}$

$$= \alpha \sum_{j=1}^{n-1} (1-\alpha)^{j-1}\, \text{SampleRTT}^{n-j} \; + (1-\alpha)^{n-1} \text{EstimatedRTT}^{1}$$

$$= \alpha \sum_{j=1}^{n-1} (1-\alpha)^{j-1}\, \text{SampleRTT}^{n-j} \quad (\because \text{EstimatedRTT}^{1} = 0)$$

（8%）

8. Consider transferring an enormous file of L bytes from Host A to Host B. Assume an MSS of 536 bytes. (a) What is the maximum value of L such that TCP sequence numbers are not exhausted? Recall that the TCP sequence number field has 4 bytes. (b) For the L you obtain in (a), find how long it takes to transmit the file. Assume that a total of 66 bytes of transport, network, and data-link header are added to each segment before the resulting packet is sent out over a 1Gbps link. Ignore flow control and congestion control so A can pump out the segments back to back and continuously. 要有說明與計算過程，以秒表示，小數點以下一位。(9%)

Ans:

There are 2^{32} = 4,294,967,296 possible sequence numbers.

a. The sequence number does not increment by one with each segment. Rather, it increments by the number of bytes of data sent. So the size of the MSS is irrelevant. The maximum size file that can be sent from A to B is simply the number of bytes representable by $\mathbf{2^{32} \approx 4.29\ \mathit{Gbytes}}$ (2%)

b. The number of segments is $\left\lceil \frac{2^{32}}{536} \right\rceil = \mathbf{8,012,999}$. (1%)

66 bytes of header get added to each segment giving a total of 66*8,012,999=528,857,934 (1%) bytes of header. The total number of bytes $\mathbf{2^{32} + 528,857,934 = 4.824 \times 10^{9}}$ bytes. (2%)
Thus it would take 4.824*8=38.6 seconds to transmit the file over a 1Gbps link. (3%)

9. Consider the two 16-bit words (shown in binary) below. Recall that to compute the Internet checksum of a set of 16-bit words, we compute the one's complement sum of the two words. That is, we add the two numbers together, making sure that any carry into the 17th bit of this initial sum is added back into the 1's place of the resulting sum); we then take the one's complement of the result. Compute the Internet checksum value for these two 16-bit words: (要寫出計算過程，8%)

10100000 10010010	this binary number is 41106 decimal (base 10)
01111110 10011111	this binary number is 32415 decimal (base 10)

Computer Networks Final (108/1)

Ans: When we add these first two numbers together, we get:

```
     10100000 10010010
     01111110 10011111      每個 bit 執行二進位加法 (4%)
   ---------------------------
   1 00011111 00110001      將進位第 17 bit 加到第一個 bit  (2%)
+                     1
   ---------------------------
     00011111 00110010
     11100000 11001101      取 1 補數 (2%)
```