

RETI DI CALCOLATORI – prova scritta del 10/09/2014

Per l'ammissione alla prova orale è necessario ottenere una valutazione sufficiente sia della prima parte che dell'intera prova.

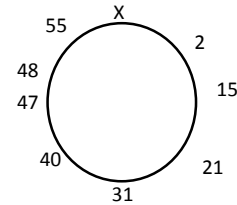
Prima parte (10 punti)

Q1. Un host A deve inviare dei dati, utilizzando TCP, a un host B presente nella stessa LAN di A. Indicare – giustificando la risposta – quanti sono tali dati se B termina di riceverli dopo 4 millisecondi nell'ipotesi che frequenza di trasmissione e ritardo di propagazione siano rispettivamente 10 Mbps e 400 microsecondi, nessun altro host collegato alla stessa LAN debba trasmettere alcunché, i preamboli di livello trasporto e rete non contengano opzioni, il valore di MSS sia 1000 byte e non si verifichino errori di trasmissione. Per semplicità, si ignorino i preamboli di livello data link e i tempi di elaborazione di A e di B.

Q2. Un host A vuole inviare un messaggio di email contenente come testo solo "OK", indirizzato a un unico destinatario. Indicare – giustificando la risposta – quanti messaggi SMTP A intercambia con il mailserver per inviare tale email.

Q3. Consideriamo una rete locale IEEE 802.11 con frequenza di trasmissione di 10 Mbps, ritardo di propagazione di 0,8 microsecondi, velocità di propagazione di $2 \cdot 10^8$ m/s, e con RTS e CTS di 20 byte, ACK di 14 byte, DIFS di 50 microsecondi e SIFS di 16 microsecondi. Supponendo che all'istante t un host A inizi il protocollo per la trasmissione di un frame di 1500 byte e che nessun altro host della rete debba trasmettere frame, indicare – giustificando la risposta – dopo quanto tempo da t un altro host può iniziare la trasmissione di frame.

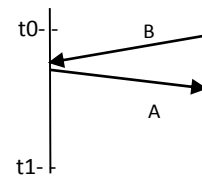
Q4. Considerare l'anello Chord a lato, che utilizza identificatori a 6 bit ed è formato dai nodi con identificatori 2, 15, 21, 31, 40, 47, 48, 55 e X, con $X \neq 55$ e $X \neq 2$. Indicare – giustificando la risposta – a quale nodo il nodo 48 inoltra la query relativa all'identificatore 57, a seconda del valore di X.



Seconda parte

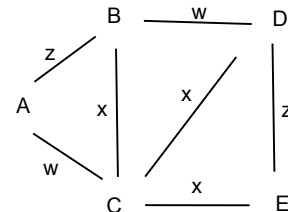
E1 (5 punti). Consideriamo una variante del protocollo Go-Back-N in cui la finestra di invio è più piccola di quella utilizzata per memorizzare i dati ricevuti dal livello superiore. Descrivere con un automa a stati finiti il comportamento del mittente di tale variante Go-Back-N utilizzando il buffer `sndpkt` per memorizzare sia i dati ricevuti dal livello superiore sia i segmenti "in volo" già spediti al destinatario. Tutte le posizioni del buffer possono essere utilizzate per memorizzare dati ricevuti dal livello superiore, mentre al più `soglia` posizioni, con `soglia < sndpkt.size()`, possono essere utilizzate per memorizzare segmenti "in volo".

E2 (5 punti). Al tempo t_0 il TCP di un processo applicativo A si trova nello stato di *slow start*, ha 3 segmenti *full-sized* in volo, il più vecchio dei quali ha numero di sequenza X, il valore di *ssthresh* è 4 MSS e deve spedire 4 MSS di nuovi dati. Indicare – giustificando la risposta – quale è il valore di *cwnd* in t_0 se, dopo avere ricevuto un riscontro non duplicato B, con $B.rwnd=3$ MSS, il TCP invia solo un segmento A *full-sized*.



Per semplicità, supporre che non scatti nessun timeout nell'intervallo $[t_0, t_1]$.

E3 (5 punti). Consideriamo la rete a lato in cui i nodi utilizzano l'algoritmo distance vector con poisoned reverse e in cui x , w e z sono interi positivi. Determinare, giustificando la risposta: (a) quali relazioni devono valere tra x , w e z affinché, quando la rete ha raggiunto lo stato di quiescenza, il numero di distanze "avvelenate" presenti nelle ultime copie dei vettori che C ha ricevuto dai suoi vicini sia massimo; (b) il contenuto del vettore delle distanze calcolato da C in tale caso.



E4 (5 punti). Supponiamo che solo tre nodi A, B e C di una rete Ethernet con topologia a bus debbano trasmettere un frame di dati. Supponiamo che al tempo t i tre nodi inizino simultaneamente a tentare di trasmettere ciascuno il proprio frame, A per la prima volta, B per la seconda e C per la terza. Indicare – giustificando la risposta – quale è la probabilità che A, B e C collidano tutti e tre insieme per 3 volte dopo t e che C riesca a trasmettere con successo il proprio frame subito dopo la terza collisione avvenuta dopo t.

Traccia della soluzione

Q1. A riceve il riscontro A del primo segmento S1 (*full sized*) contenente dati dopo

$$\left(\frac{(20+20) \times 8}{10^7} + \frac{400}{10^6} + \frac{(20+20) \times 8}{10^7} + \frac{400}{10^6} + \frac{(20+20+1000) \times 8}{10^7} + \frac{400}{10^6} + \frac{(20+20) \times 8}{10^7} + \frac{400}{10^6} \right) s = 2528 \mu s$$

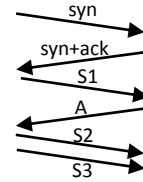
Quindi B termina di ricevere il secondo (S2) e il terzo segmento (S3) contenente dati dopo

$$\left(2528 + \frac{(20+20+1000) \times 8}{10} + X + 400 \right) \mu s$$

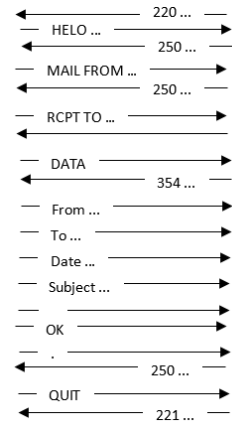
dove X è il ritardo di trasmissione di S3. Quindi X = 240 μs e S3 contiene D byte di dati dove

$$\frac{(20 + 20 + D) \times 8}{10^7} = \frac{240}{10^6}$$

ovvero D=260. Quindi i byte di dati inviati da A sono in totale 1000+1000+260=2260.



Q2. In totale 19 messaggi SMTP¹: 3 per apertura connessione + 4 per busta + 6 per intestazioni + 1 per linea vuota + 3 per corpo + 2 per chiusura connessione.

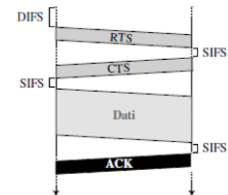


Q3. Il canale sarà di nuovo libero dopo

$$DIFS + d_{tras}(RTS) + d_{prop} + SIFS + d_{tras}(CTS) + d_{prop} + SIFS + d_{tras}(Dati) + d_{prop} + SIFS + d_{tras}(ACK) + d_{prop} = (50+16+0,8+16+16+0,8+16+1200+0,8+16+11,2+0,8) \mu s = 1344,4 \mu s.$$

Quindi un altro host potrà iniziare la trasmissione di un frame dopo

$$(1344,4 + 50+16+0,8+16+16+0,8+16) \mu s = 1460 \mu s.$$



Q4. Analizziamo i vari casi possibili:

(a) se X=56 i finger del nodo 48 sono

55	55	55	X	2	21
----	----	----	---	---	----

(b) se X∈[57,63] i finger del nodo 48 sono

55	55	55	X	2	21
----	----	----	---	---	----

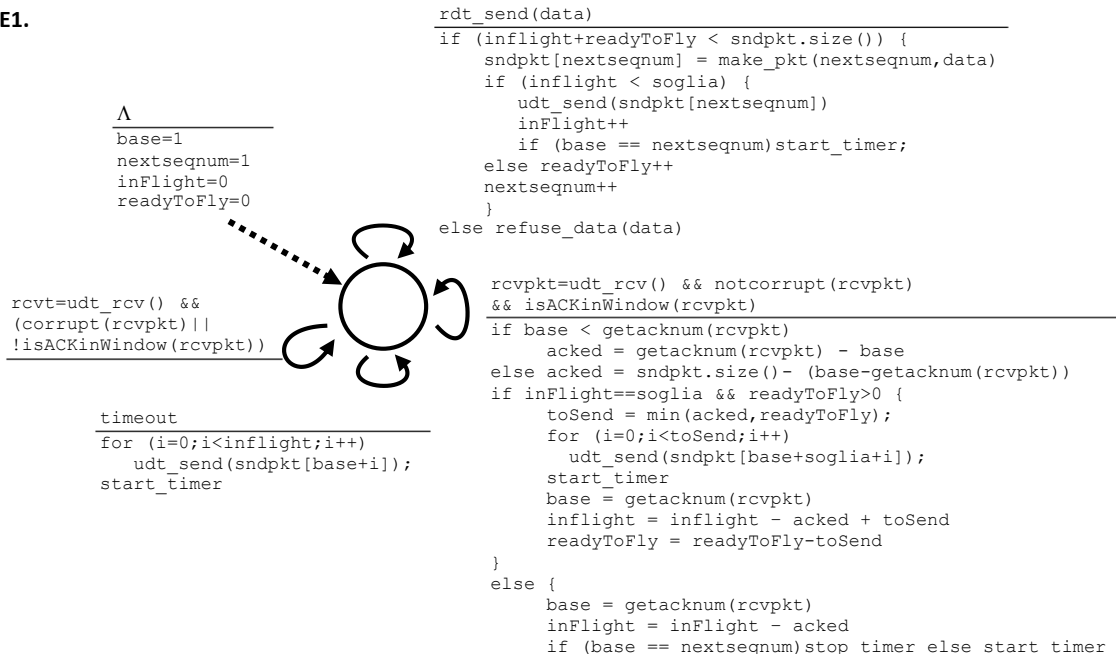
(c) se X∈[0,1] i finger del nodo 48 sono

55	55	55	X	X	21
----	----	----	---	---	----

Il nodo 48 inoltra quindi la query relativa all'identificatore 57 al nodo X nel caso (a), ovvero se X=56, inoltra invece la query al nodo 55 nei casi (b) e (c), ovvero se X∈[57,63] o X∈[0,1].

¹ 14 se non vengono inviati i messaggi opzionali contenenti le intestazioni e la linea vuota.

E1.



E2. Analizziamo i vari casi possibili.

- Se $B.ackN = X+3$ MSS allora, dopo avere ricevuto B, A non avrebbe più dati in volo e quindi $\min(cwnd_{t_0}+1MSS, 3MSS)$ dovrebbe essere 1 MSS. Ciò non è possibile poiché la dimensione di $cwnd$ è sempre almeno 1 MSS.
- Se $B.ackN = X+2$ MSS allora, dopo avere ricevuto B, A ha ancora 1 MSS di dati in volo e quindi $\min(cwnd_{t_0}+1MSS, 3MSS)$ deve essere 2 MSS, ovvero $cwnd_{t_0}$ doveva essere 1 MSS.
- Se $B.ackN = X+1$ MSS allora, dopo avere ricevuto B, A ha ancora 2 MSS di dati in volo e quindi $\min(cwnd_{t_0}+1MSS, 3MSS)$ deve essere 3 MSS, ovvero $cwnd_{t_0}$ doveva essere almeno 2 MSS e meno di 4 MSS (altrimenti il TCP in t_0 non si sarebbe trovato nello stato di slow start).

E3. (a) I vettori ricevuti da C conterranno il minimo dei valori indicati in ogni casella della tabella sottostante, dove i valori evidenziati in grassetto causerebbero l'avvelenamento della distanza. Affinché il numero di distanze "avvelenate" presenti in tali vettori sia massimo deve quindi valere: $x+w < z$, $2x < w$ e $2x < z$.

	A	B	D	E
A	-	$\min(x+w, z)$	$\min(x+w, 2x+z, w+2z)$	$\min(x+w, 2x+z, w+2z)$
B	$\min(x+w, z)$	-	$\min(2x, w)$	$\min(2x, w+z)$
D	$\min(x+w, w+z, 2x+z)$	$\min(2x, w)$	-	$\min(2x, z)$
E	$\min(x+w, w+2z, 2x+z)$	$\min(2x, w+z)$	$\min(2x, z, x+2w+z)$	-

A	w
B	x
D	x
E	x

E4. Affinché i tre nodi collidano tutti e tre insieme per tre volte dopo t , essi devono attendere lo stesso $K \times T_{fr}$ tempo sia dopo la prima che dopo la seconda collisione avvenuta dopo t . Dopo la terza collisione C deve invece scegliere di trasmettere subito ($K=0$), e A e B di non farlo ($K \neq 0$). Tutto ciò può avvenire con probabilità $\left(2 \times \frac{1}{2} \times \frac{1}{4} \times \frac{1}{8}\right) \times \left(4 \times \frac{1}{4} \times \frac{1}{8} \times \frac{1}{16}\right) \times \left(\frac{7}{8} \times \frac{15}{16} \times \frac{1}{32}\right) = \frac{105}{2^{24}}$.