

## RETI DI CALCOLATORI – Sesto appello, a.a. 2009/2010

La prova è strutturata in due parti. Per ottenere una valutazione sufficiente dell'intera prova è necessario ottenere una valutazione sufficiente della prima parte. In accordo con quanto deliberato dal Consiglio di Facoltà il 15.12.2009, ogni studente può partecipare a tutti i cinque appelli previsti e consegnare al più quattro prove scritte.

### Prima parte (10 punti)

**Q1.** Supponiamo che un router A trasmetta un pacchetto sul collegamento col router B e che il secondo bit del pacchetto venga immesso sul collegamento nel momento in cui il primo bit si trova a metà del collegamento. Determinare – giustificando la risposta – la lunghezza del collegamento supponendo che la velocità di trasmissione del collegamento sia 1 Mbps e che la velocità di propagazione sia  $2 \cdot 10^8$  m/s.

**Q2.** Indicare – giustificando la risposta – se è possibile o meno che il TCP di un processo applicativo A abbia una quantità di dati “in volo” (ovvero spediti ma non ancora riscontrati) maggiore dell'ultimo valore della dimensione della finestra di ricezione che ha ricevuto dal suo pari.

**Q3.** Sia A un router che utilizza il protocollo distance vector e supponiamo che il vettore delle distanze di A contenga  $D_A(V1)=x$  e  $D_A(V2)=y$ , dove V1 e V2 sono gli unici due router con cui A ha un collegamento diretto. Supponiamo adesso che A riceva il vettore delle distanze di V1 contenente  $D_{V1}(Z)=y$  e dopo quello di V2 contenente  $D_{V2}(Z)=x+1$ . Indicare – giustificando la risposta – quale è la distanza  $D_A(Z)$  che A avrà determinato dopo avere ricevuto i due suddetti vettori.

**Q4.** Indicare quale è l'obiettivo dell'algoritmo MD5.

### Seconda parte

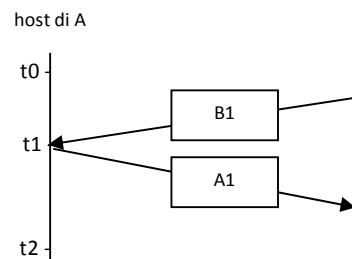
**E1 (6 punti).** Descrivere con un automa a stati finiti il comportamento di un cliente che, ricevuta una richiesta `resolve(nomeHost)` di un'applicazione, restituisce l'indirizzo IP dell'host specificato e –se esso si trova sulla stessa rete– anche il corrispondente indirizzo fisico. Per semplicità assumiamo che nessun messaggio DNS vada perso, che le query DNS vengano risolte in modo ricorsivo e che il cliente invii al più due volte ciascuna query ARP prima di stabilire che l'host non appartiene alla rete locale. Nella descrizione dell'automa indicare

con	<code>UDPSend(IPserver, porta, &lt;QNAME, QTYPE&gt;)</code>	l'invio di una richiesta DNS con UDP,
con	<code>... = UDPreceive()</code>	la ricezione di una risposta DNS,
con	<code>... = ARPcache(IPaddress)</code>	la consultazione della cache ARP,
e con	<code>bcast(ARPquery)</code>	l'invio di una query ARP in broadcast.

**E2 (6 punti).** Supponiamo che al tempo  $t_0$  il TCP di un processo applicativo A abbia 2 MSS di dati in volo con  $\text{SendBase}=X+3\text{MSS}$ , che la dimensione della sua finestra di congestione sia 1 MSS, che si trovi nello stato di slow start e che abbia altri 2 MSS di nuovi dati da spedire. Supponendo che B1 sia un riscontro che non contiene dati, che nell'intervallo  $[t_0, t_2]$  A non riceva né spedisca altri segmenti e che non scada nessun timeout per A, indicare – giustificando la risposta:

- i possibili valori di `ackNum` e di `RcvWin` contenuti in B1,
- i possibili valori di `seqNum` e la quantità di dati contenuti in A1,
- i possibili valori di `CongWin`, `SendBase` e `NextSeqNum` in  $t_2$ .

Per semplicità supponiamo che tutti i segmenti scambiati da A e B contenenti dati contengano 1 MSS di dati.



**E3 (4 punti).** Supponiamo che un router R di un sistema autonomo che utilizza RIP memorizzi solo la sua tabella di instradamento utilizzando un vettore `TR` dove `TR[d].next` e `TR[d].hops` indicano rispettivamente il prossimo hop e il numero di hop per raggiungere la destinazione d. Descrivere con uno pseudocodice il modo in cui R aggiorna la sua tabella di instradamento quando riceve un advertisement `TV` da un suo vicino V.

**E4 (4 punti).** Consideriamo una rete Ethernet in cui solo quattro nodi devono spedire dati. Supponendo che tutti e quattro i nodi inizino simultaneamente a trasmettere e che quindi collidano, indicare –giustificando la risposta– quale è la probabilità che collidano tutti e quattro di nuovo insieme nei due successivi tentativi di spedizione che effettueranno.

## TRACCIA DELLA SOLUZIONE

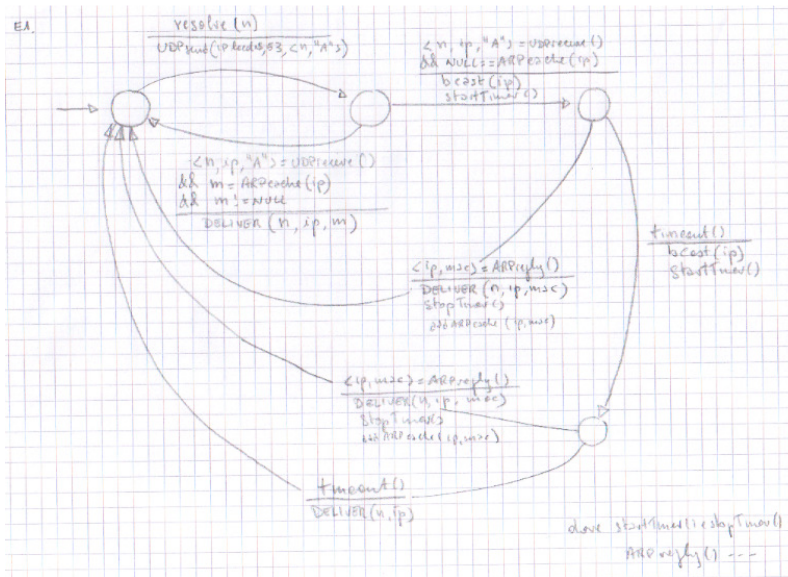
**Q1.** Se il secondo bit del pacchetto viene immesso sul collegamento nel momento in cui il primo bit del pacchetto si trova a metà del collegamento significa che  $2/R = 1/R + d_{prop}/2$  dove  $R$  è la frequenza di trasmissione e  $d_{prop}$  il ritardo di propagazione. Abbiamo quindi che  $d_{prop} = 2/R$  e, dato che  $d_{prop}$  è dato dalla lunghezza  $d$  del collegamento diviso la velocità di propagazione, otteniamo:

$$d = 2/10^6 \times 2 \times 10^8 \text{ m} = 400 \text{ m}.$$

**Q2.** Sì, ciò avviene ogni volta che l'host di A, dopo avere ricevuto un riscontro contenente  $RcvWin=0$ , invia un segmento di "zero widow probe" contenente un byte di nuovi dati.

**Q3.**  $D_A(Z) = \min(C(A, V1) + y, C(A, V2) + x + 1)$  -dove  $C(A, V1)$  e  $C(A, V2)$  sono i costi dei collegamenti da A a V1 e da A a V2 rispettivamente.

**Q4.** L'obiettivo dell'algoritmo MD5 è realizzare una funzione hash crittografica, ovvero una funzione  $H$  in grado di "triturare" un messaggio  $m$  in una stringa  $H(m)$  di lunghezza fissata in modo che sia computazionalmente impossibile trovare un messaggio  $m'$  diverso da  $m$  tale che  $H(m') = H(m)$ . MD5 viene utilizzata per verificare l'integrità dei messaggi e nelle firme digitali.



**E2.** Distinguiamo tre casi possibili:

- B1 contiene un riscontro duplicato ricevuto per la terza volta dall'host di A, ovvero  $B1.ackNum = X + 3MSS$  (il valore di  $B1.RcvWin$  è irrilevante). In questo caso per il meccanismo della ritrasmissione veloce A1 sarà la ritrasmissione del più vecchio pacchetto ancora in volo, ovvero  $A1.seqNum = X + 3MSS$  e A1 conterrà 1 MSS di dati. I valori di  $CongWin$ ,  $SendBase$  e  $NextSeqNum$  in  $t2$  rimarranno quindi 1MSS,  $X + 3MSS$  e  $X + 5MSS$  rispettivamente.
- B1 contiene un riscontro per il pacchetto più vecchio in volo,  $B1.ackNum = X + 4MSS$ . In questo caso  $CongWin$  diventerà 2MSS e  $B1.RcvWin$  dovrà essere  $\geq 2MSS$  dato che l'host di A spedisce nuovi dati,  $A1.seqNum = X + 5MSS$  e A1 conterrà 1 MSS di dati. I valori di  $CongWin$ ,  $SendBase$  e  $NextSeqNum$  in  $t2$  saranno quindi 2MSS,  $X + 4MSS$  e  $X + 6MSS$  rispettivamente.
- B1 contiene un riscontro per il pacchetto più giovane in volo,  $B1.ackNum = X + 5MSS$ . In questo caso  $CongWin$  diventerà 2MSS e  $B1.RcvWin$  dovrà essere 1MSS dato che l'host di A spedisce 1 solo MSS di nuovi dati,  $A1.seqNum = X + 5MSS$  e A1 conterrà 1 MSS di dati. I valori di  $CongWin$ ,  $SendBase$  e  $NextSeqNum$  in  $t2$  saranno quindi 2MSS,  $X + 5MSS$  e  $X + 6MSS$  rispettivamente.

**E3.** /\* Assumiamo per semplicità che le destinazioni siano rappresentate da interi. \*/

forEach destinazione  $d$  in  $TV$ :

if ( $TR[d].hops > 1 + TV[d].hops$ ) || ( $TR[d].next == V$ )

then { $TR[d].hops = 1 + TV[d].hops$ ;

$TR[d].next = V$ ; }

**E4.** Dopo che si è verificata la prima collisione ciascuno nodo attenderà  $K \cdot 512$ bit, con  $K$  valore intero scelto in modo casuale nell'intervallo  $[0, 2^1 - 1]$ , prima di tentare nuovamente la spedizione. La probabilità che collidano tutti e quattro di nuovo al secondo tentativo è quindi  $2/2^4$  (ovvero la probabilità che tutti e quattro scelgano 0 oppure 1). In modo del tutto analogo la probabilità che collidano tutti e quattro di nuovo anche al terzo tentativo è  $4/4^4$ . La probabilità che collidano tutti e quattro di nuovo nei due successivi tentativi è quindi  $2/2^4 \times 4/4^4 = 1/512$ .