

Reti di calcolatori

Prova scritta – 24 luglio 2012

Parte 1

Esercizio 1.1 [12 punti]

Si vuole realizzare un servizio Web per la raccolta delle iscrizioni di atleti a gare di nuoto. Il servizio prevede le seguenti pagine:

1. Un form in cui l'utente indica il nominativo in un campo di testo, le gare a cui intende iscriversi in un menu a scelta multipla, la gara su cui ha ottenuto la qualificazione in un menu a scelta singola e il tempo corrispondente, espresso in secondi e frazioni di secondo (esempio 58.34), ottenuto sulla gara di qualificazione (si usi un campo testo). I due menu devono essere generati dinamicamente a partire da un array che contiene la lista delle gare (si ipotizzi ad esempio la lista "100m SL", "100m rana", "200m SL", "200m rana", "4x100m misti").
2. Una pagina di raccolta dei dati che memorizza sul server gli inserimenti fatti col form di cui al punto 1 nella stessa sessione di lavoro. Se si inserisce più volte uno stesso atleta, si aggiornano le opzioni già inserite in precedenza, ovvero le gare a cui richiede di partecipare sono aggiunte a quelle già inserite e la gara di qualificazione viene aggiunta se non è stata inserita in precedenza o ne viene sovrascritto il tempo indicato se è già stata inserita.
3. Una pagina di riepilogo che stampa una tabella con il numero di iscrizione per gara. Si riporta poi per ogni gara la lista di tutti gli atleti che hanno il tempo di qualificazione migliore su tale gara.

Si scrivano il form e le due pagine di raccolta dati e riepilogo usando HTML/PHP. Si supponga di utilizzare il metodo POST nel form.

Esercizio 1.2 [4 punti]

Dato il seguente codice PHP

```
$v1 = 1;  
$v2 = 2.3;  
$a = array("b$v1" => '11$v1', "c$v1" => "2+$v2");  
$res = $a["b1"]%$a["c1"];
```

indicare quali sono le variabili definite nell'interprete dopo l'esecuzione del codice, con il loro tipo e valore.

Domanda 1 [7 punti]

Spiegare i meccanismi POST e GET con cui vengono inviati dati da un form Web ad un'applicazione server-side mettendo in evidenza i pro e i contro delle due soluzioni.

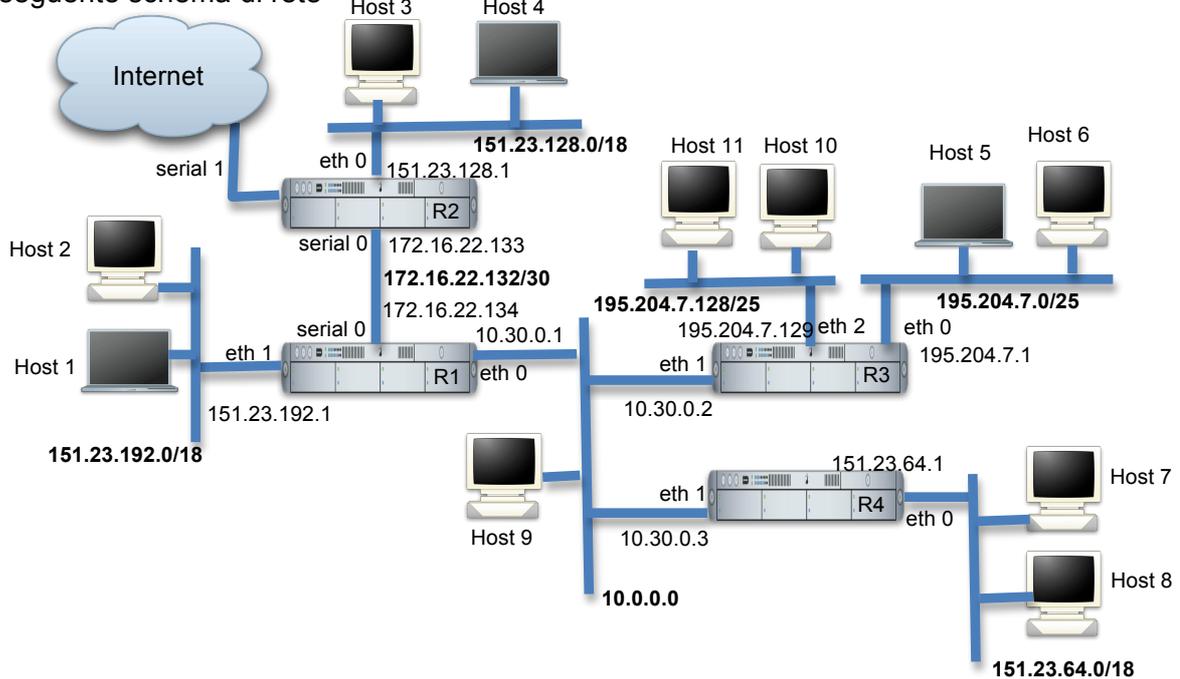
Domanda 2 [7 punti] (solo per chi deve recuperare la parte 1)

Indicare a quali livelli di un'architettura di rete può essere necessario definire un meccanismo di indirizzamento indicandone le funzionalità.

Parte 2

Esercizio 2.1 [punti 6]

Dato il seguente schema di rete



scegliere la configurazione di rete dell'host 11 (IP, netmask e configurazione di routing) e indicare il contenuto delle tabelle di routing del router R1.

Esercizio 2.2 [punti 8]

Si vuole definire un formato per l'interscambio dati relativo ai dati ospedalieri di un insieme di pazienti. Il file scambiato specifica la lista dei pazienti. Per ciascun paziente si indica il nominativo (string), il codice fiscale (string), la data di nascita (date) e la lista dei ricoveri. Per ogni ricovero si specificano la data di inizio (date), il numero di giorni (integer) e la descrizione delle cure (string). Si proponga la struttura XML necessaria, mostrando un esempio, e si scriva il file XML schema associato.

Esercizio 2.3 [punti 8]

Si scriva il codice javascript necessario ad implementare la seguente funzionalità in una pagina HTML: Entrando in una cella di una tabella identificata dall'id "imgbox", si cambia l'immagine mostrata sullo sfondo scorrendo circolarmente un vettore che contiene i nomi delle immagini disponibili (l'immagine sullo sfondo è definita dalla proprietà di stile background-image a cui si associa come valore l'url del file immagine – es. background-image: url('img.jpg')).

Esercizio 2.4 [punti 8]

Supponendo che un socket client UDP sia già stato opportunamente inizializzato e sia disponibile il suo identificatore nella variabile `sd` (`int sd`) e che la variabile `struct sockaddr_in sad` sia già stata inizializzata con l'indirizzo del server, scrivere il codice per implementare il seguente protocollo di comunicazione.

1. Il client chiama la funzione predefinita `char *getQuery()` e poi invia un pacchetto contenente la stringa ottenuta concatenando la stringa "QRY?" con il risultato della chiamata della funzione predefinita.
2. Il client attende un pacchetto di risposta dal server.
3. Fino a che il pacchetto ricevuto contiene la stringa "RTR" il client ripete l'invio della stringa.
4. Altrimenti chiama la funzione `int setReply(char *msg)` passando come parametro la stringa ricevuta dal server e ripete la sequenza dal passo 1.

Reti di calcolatori

Prova scritta – 24 luglio 2012

Parte 1

Esercizio 1.1 [12 punti]

Si vuole realizzare un servizio Web per la raccolta di offerte di acquisto per l'asta di una serie di prodotti. Il servizio prevede le seguenti pagine:

4. Un form in cui l'utente indica il nominativo in un campo di testo, i prodotti a cui è interessato in un menu a scelta multipla, il prodotto per il quale fa una specifica offerta di acquisto in un menu a scelta singola e l'ammontare dell'offerta, espresso in euro e centesimi di euro (esempio 67.56) in un campo testo. I due menu devono essere generati dinamicamente a partire da un array che contiene la lista dei prodotti (si ipotizzi ad esempio la lista "bulloni", "viti", "dadi", "chiodi", "rivette").
5. Una pagina di raccolta dei dati che memorizza sul server gli inserimenti fatti col form di cui al punto 1 nella stessa sessione di lavoro. Se si inserisce più volte uno stesso utente, si aggiornano le opzioni già inserite in precedenza, ovvero i prodotti a cui è interessato sono aggiunti a quelli già inseriti e l'offerta specifica per un prodotto viene aggiunta se non è stata inserita in precedenza o ne viene sovrascritto l'ammontare se è già stata inserita.
6. Una pagina di riepilogo che stampa una tabella con il numero di utenti interessati per prodotto. Si riporta poi per ogni prodotto la lista di tutti gli utenti che hanno fatto l'offerta migliore per tale prodotto.

Si scrivano il form e le due pagine di raccolta dati e riepilogo usando HTML/PHP. Si supponga di utilizzare il metodo POST nel form.

Esercizio 1.2 [4 punti]

Dato il seguente codice PHP

```
$v1 = "2";  
$v2 = 22.3;  
$a = array("z$v1" => "$v2*2", "y$v1" => '2*$v1');  
$res = $a["z2"]*$a["y2"];
```

indicare quali sono le variabili definite nell'interprete dopo l'esecuzione del codice, con il loro tipo e valore.

Domanda 1 [7 punti]

Spiegare come il protocollo HTTP supporta lo scambio di una risorsa permettendo l'interpretazione corretta da parte del client in base al suo formato dati.

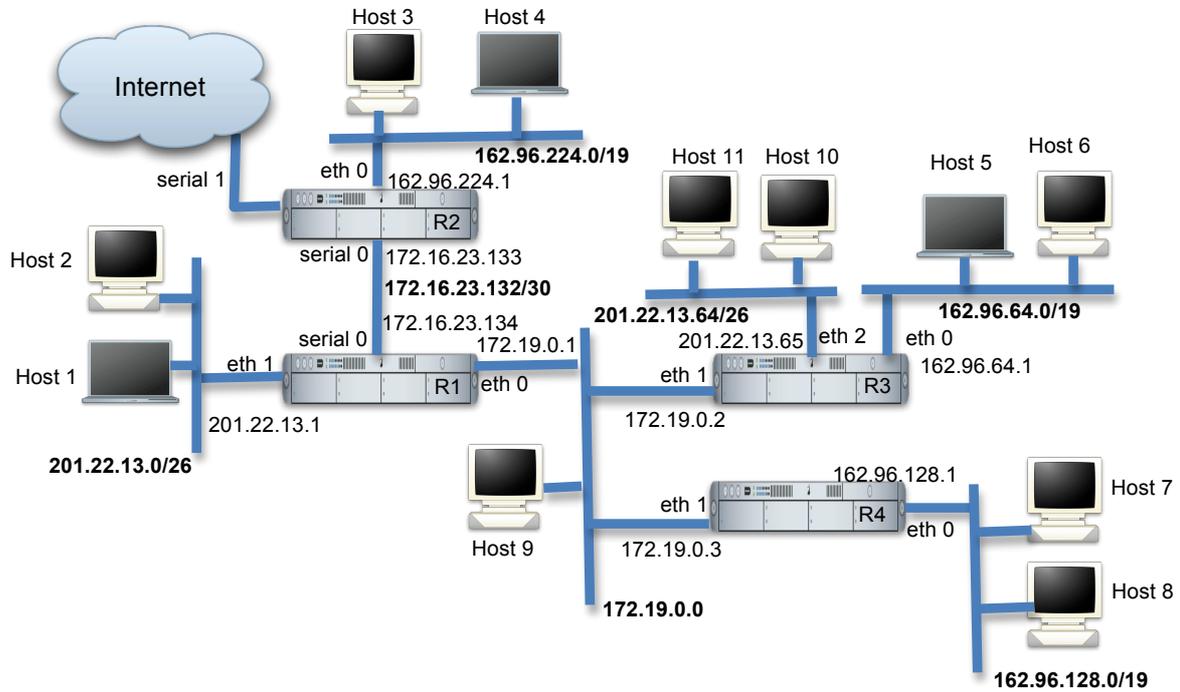
Domanda 2 [7 punti] (solo per chi deve recuperare la parte 1)

Illustrare la struttura gerarchica dello spazio dei nomi DNS indicando il ruolo dei server di competenza di zona nella risoluzione dei nomi.

Parte 2

Esercizio 2.1 [punti 6]

Dato il seguente schema di rete



scegliere la configurazione di rete dell'host 6 (IP, netmask e configurazione di routing) e indicare il contenuto delle tabelle di routing del router R1.

Esercizio 2.2 [punti 8]

Si vuole definire un formato per l'interscambio dati relativo ai dati di installazione del software per i computer di un'azienda. Il file scambiato specifica la lista dei computer. Per ciascun computer si indica l'indirizzo IP (string), la descrizione (string), la data di acquisto (date) e la lista dei software installati. Per ogni software si specificano il nome del software (string), la versione (integer) e la data di installazione (date). Si proponga la struttura XML necessaria, mostrando un esempio, e si scriva il file XML schema associato.

Esercizio 2.3 [punti 8]

Si scriva il codice javascript necessario ad implementare la seguente funzionalità in una pagina HTML: Entrando in una cella di una tabella identificata dall'id "msgbox", si cambia il testo contenuto scorrendo circolarmente un vettore che contiene le stringhe dei messaggi predefiniti.

Esercizio 2.4 [punti 8]

Supponendo che un socket client TCP sia già stato opportunamente inizializzato (il suo identificatore è disponibile nella variabile `sd (int sd)`) e sia già avvenuta la connessione con il server, scrivere il codice per implementare il seguente protocollo di comunicazione.

1. Il client chiama la funzione predefinita `char *newCmd()` e poi invia al server la stringa ottenuta concatenando la stringa "CMD=" con il risultato della chiamata della funzione.
2. Il client attende una risposta dal server.
3. Fino a che la risposta è la stringa "FAIL" il client ripete l'invio della stringa.
4. Altrimenti chiama la funzione `int opResult(char *res)` passando come parametro la stringa ricevuta dal server e ripete la sequenza dal passo 1.