

Reti di calcolatori

Prova scritta – 18 giugno 2012

Parte 1

Esercizio 1.1 [12 punti]

Si vuole realizzare un servizio Web per la raccolta delle iscrizioni ad un insieme di scuole. Il servizio prevede le seguenti pagine:

1. Un form in cui l'utente può scegliere le scuole in un menù a scelta multipla e inserire il nome della persona in un campo testuale. Il menù a scelta multipla per le scuole deve essere generato dinamicamente da un array (si considerino ad esempio le scuole "S. Genoveffa", "S. Egidia", "S. Ermanna", "S. Gaspare"). Il form permette inoltre di specificare la fascia di reddito e il tipo di impiego dei genitori. Queste opzioni devono essere specificate da due menù a tendina generati dinamicamente. Ad ogni opzione dei due menù è associato un valore di punteggio. Si possono assumere come esempio i seguenti valori memorizzati in due array: per le fasce di reddito "<2500" => 4, "<7500" => 3, "<12500" => 2, "<17500" => 1, ">20000" => 0 ; per il tipo di impiego "entrambi TP" => 6, "uno TP, uno MT" => 4, "monogenitore" => 6, "monogenitore TP" => 10, "uno non occupato" => 0.
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. Per ogni alunno non importa mantenere le opzioni di reddito e impiego dei genitori ma è sufficiente memorizzare il punteggio totale associato alle opzioni selezionate. Se si inserisce più volte uno stesso alunno, si sovrascrivono le richieste già fatte.
3. Una pagina di riepilogo che stampa una tabella con il numero di richieste per scuola. Si riporta poi per ogni scuola la lista di tutti i richiedenti che ha il punteggio massimo per quella scuola.

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 = 10.2;  
$v2 = "33";  
$a = array('v1' => "1$v1", 'v2' => '1$v2');  
$res = $a["v1"]+$a["v2"];
```

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

Domanda 1 [7 punti]

Motivare l'uso di cache e proxy Web illustrando il supporto fornito dal protocollo HTTP/1.1 per la loro gestione.

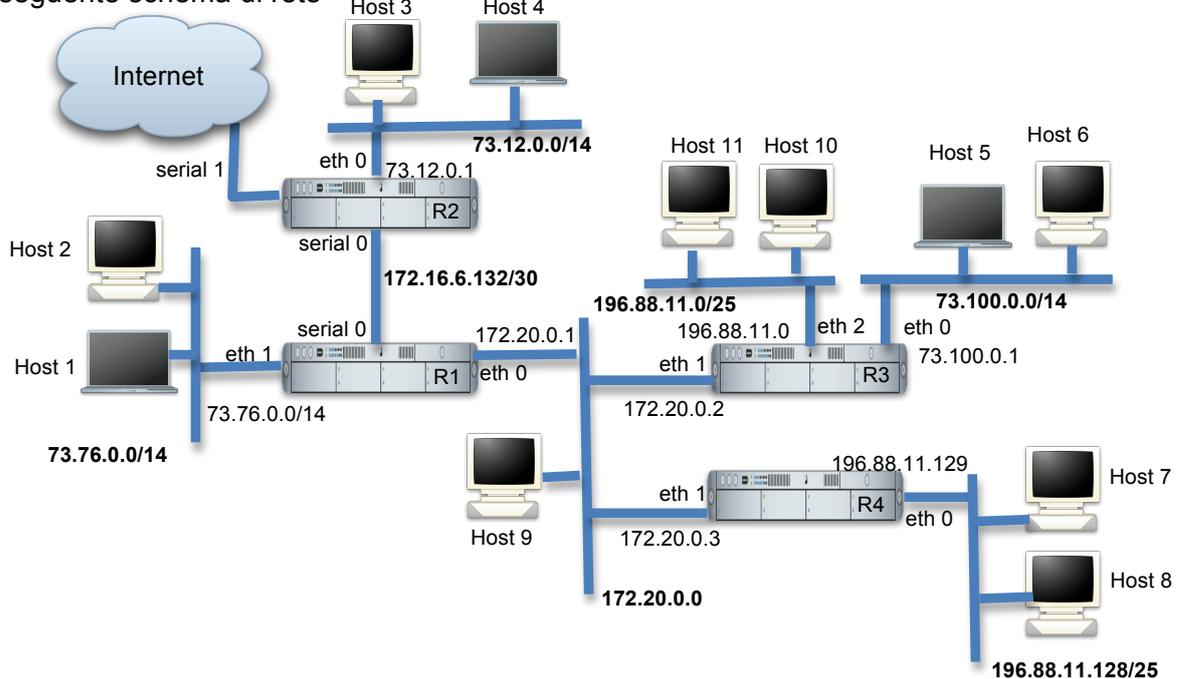
Domanda 2 [7 punti] (solo per chi deve recuperare la parte 1)

Illustrare il meccanismo di interrogazione iterativa nel sistema DNS specificando le tipologie di server coinvolti.

Parte 2

Esercizio 2.1 [punti 6]

Dato il seguente schema di rete



scegliere la configurazione di rete dell'host 8 (IP, netmask e configurazione di routing) e indicare il contenuto delle tabelle di routing del router R3.

Esercizio 2.2 [punti 8]

Si vuole definire un formato per l'interscambio dati relativo all'inventario dei beni di un dipartimento. Il file scambiato specifica la lista degli affidatari e la lista dei beni inventariati. Per ciascun affidatario si indica il nominativo (string), la matricola (integer) e l'ufficio di appartenenza (string). Per ciascun bene inventariato si specificano il codice (integer), la descrizione (string) e la matricola dell'affidatario (integer). 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: Inserendo un testo in un elemento textarea (elemento HTML `<textarea>`), ogni parola inserita (il separatore è lo spazio codice 32) viene aggiunta ad un elemento (tag HTML ``) ad una lista predefinita (tag HTML ``) nel file HTML.

Esercizio 2.4 [punti 8]

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

1. Dopo aver accettato una connessione, il server attende una stringa dal client.
2. Chiama la funzione `char *getKey(char *msg)` passando come parametro la stringa ricevuta e invia al client la stringa ottenuta dalla funzione.
3. Attende una nuova stringa da parte del client. Se la stringa ricevuta inizia con "KEY" invia la client la stringa ottenuta dalla funzione `char *getValue(char *key)` passando come parametro la stringa ricevuta a partire dal primo carattere dopo la sottostringa "KEY" e poi chiude la connessione e si rimette in attesa di altre richieste di connessione. Altrimenti ripete lo scambio dal punto 2 utilizzando la stringa ricevuta come nuovo parametro della funzione `getKey()`.