Nome Cognome Matricola_____

Reti di calcolatori

Prova scritta – 6 febbraio 2013

Esercizio 1 [16 punti]
Si vuole realizzare un servizio Web per la raccolta delle iscrizioni degli utenti di una palestra ai turni delle attività di allenamento. Il servizio prevede le seguenti pagine:

1. Un form in cui l'utente può scegliere l'attività di allenamento a cui si vuole iscrivere un menù a scelta singola, i turni scelti da un menù a scelta multipla e inserire il proprio nome. Il form permette di specificare un insieme di turni per una data attività ad ogni invio. Le opzioni dei menù devono essere generate dinamicamente a partire da array. Si assuma un array che contiene la lista ("pilates", "aerobica", "step", "spinning", "corpo libero") per le attività e ("lunedì", "martedì", "mercoledì", "giovedì", "venerdì") per i turni.
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. Gli inserimenti fatti dallo stesso utente si uniscono a quelli fatti eventualmente in precedenza.
3. Una pagina di riepilogo che stampa una tabella con il numero di utenti registrati per ogni opzione attività-turno. Per ogni attività si riportano poi tutti i turni che hanno ottenuto il massimo delle registrazioni e si stampano i nomi degli utenti iscritti.

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 2 [5 punti]
Dato il seguente codice PHP

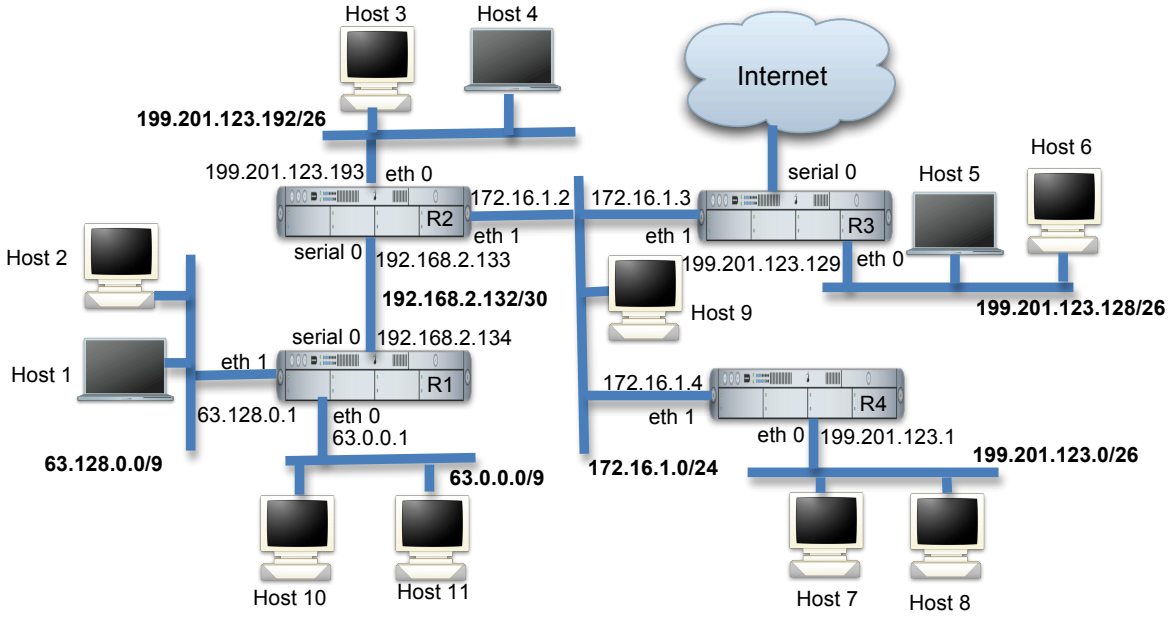
```
$a = "1.2";
$b = "2$a";
$v = array('x'=>$a,'y'=>$b,'z'=>'3$a');
$sum = $v['x']+$v['y']+$v['z'];
```

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

Domanda 1 [9 punti]
Indicare come gli agenti coinvolti nel processo di invio, instradamento, ricezione e visualizzazione dei messaggi di email utilizzano i principali campi di intestazione in un messaggio.

Nome Cognome Matricola_____

Esercizio 3 [punti 6]
Dato il seguente schema di rete



scegliere la configurazione di rete dell'host 3 (IP, netmask e configurazione di routing) e indicare il contenuto delle tabelle di routing del router R2.

Esercizio 4 [punti 8]
Si vuole definire un formato per l'interscambio dati relativo alle chiamate di soccorso di una centrale operativa. *Il file scambiato specifica la lista delle chiamate. Ciascuna chiamata è caratterizzata da un timestamp (dateTime), dal numero di telefono da cui è stata ricevuta (string), dal codice dell'operatore (integer). Alla chiamata è inoltre associata una lista di segnalazioni. Per ciascuna segnalazione è specificato il luogo (string), la descrizione (string) e il livello di rischio (integer).* Si proponga la struttura XML necessaria, mostrando un esempio, e si scriva il file XML schema associato.

Esercizio 5 [punti 8]
Si scriva il codice javascript necessario ad implementare la seguente funzionalità in una pagina HTML: *cambiando la selezione in un insieme di radio button (elementi <INPUT type="radio">) predefinito (attributo name assegnato), a ciascuno dei quali è associato un valore di colore (es. 4 elementi con valori "red", "green", "blue", "yellow"), si cambia in modo corrispondente il colore dei caratteri delle intestazioni di terzo livello (elementi <H3>). Suggerimenti: il cambiamento della selezione di un radio button genera un evento di tipo "change" e la libreria DOM prevede il metodo getElementsByName.*

Esercizio 6 [punti 8]
Supponendo che il riferimento a un socket server UDP sia già disponibile nella variabile `int sd` scrivere il codice C per implementare il seguente protocollo di comunicazione.

1. Il server attende un pacchetto da un client.
2. Se i dati del pacchetto iniziano con la sequenza di caratteri "STO" si chiama la funzione `char *storeBlock(void *blockPtr)` passandogli come parametro il puntatore al quarto byte del pacchetto. La funzione fornisce una stringa che viene inviata client.
3. Se il pacchetto inizia con la stringa "LD" si invia al client un pacchetto di risposta con 1Kbyte di dati ottenuti dalla chiamata della funzione funzione `void *loadBlock(char *id)`, passando come parametro id il puntatore al carattere che segue "LD" nel pacchetto ricevuto.
4. In tutti gli altri casi non si risponde.
5. Dopo aver analizzato il pacchetto ricevuto si ritorna al punto 1