

Corso di Tecnologie Web AA 2005/2006
14/3/2006

L'esame del corso di Tecnologie Web consiste nello svolgimento di un progetto o di una tesina, e in una prova scritta su tutti gli argomenti discussi nel corso. Il voto del progetto/tesina pesa per il 60% sul voto finale dell'esame, mentre il voto della prova scritta pesa per il rimanente 40%. Si supera l'esame del corso (e si può quindi registrare il voto) se $0.6 * \text{voto_progetto} + 0.4 * \text{voto_scritto} \geq 18$.

I progetti o le tesine devono essere tassativamente consegnati (è possibile consegnare via mail) entro la data indicata. Saranno resi disponibili nuovi progetti e nuove tesine per ogni sessione d'esame. Ciascuno studente è libero di scegliere di svolgere un progetto oppure una tesina; l'argomento va scelto tra quelli indicati dal docente.

I progetti e le tesine sono descritti in modo volutamente generico. Vengono indicati i contenuti minimali che devono essere realizzati; siete ovviamente liberi (e incoraggiati) ad espandere la portata del vostro lavoro per includere anche aspetti non espressamente richiesti nel testo. Non esagerate! Questi sono progetti da completare in tempi ragionevoli e senza eccessivi sforzi. Un livello di sofisticazione eccessivo è inutile.

I progetti riguardano lo sviluppo di semplici applicazioni (tipicamente, ma non necessariamente, in un linguaggio a scelta tra C/C++ e Java). Dei progetti va consegnato un archivio (in formato .zip o .tar compresso) contenente almeno i sorgenti commentati dell'applicazione, unitamente ad una documentazione cartacea che spieghi la struttura generale dell'applicazione, giustifichi le decisioni implementative fatte, e includa un manuale utente che spieghi come compilare e utilizzare l'applicazione.

Le tesine riguardano approfondimenti su argomenti trattati a lezione. La tesina va consegnata preferibilmente in formato PDF o Postscript compresso, oppure in formato cartaceo entro la scadenza. Le tesine devono includere una bibliografia che elenchi gli eventuali documenti consultati per approfondire l'argomento scelto.

I progetti e le tesine devono essere lavoro originale dello studente. Copiare il codice o il contenuto della tesina da qualunque fonte comporterà l'annullamento della prova.

I progetti e le tesine devono essere consegnati entro <u>martedì 4 aprile 2006</u>

Progetti:

1. Verificare automaticamente l'accessibilità dei siti WEB (<http://www.w3.org/TR/WCAG10/>) è in generale molto più difficile che valutarne ad esempio la correttezza sintattica rispetto ad un DTD. Non esistono attualmente dei sistemi totalmente automatici per la verifica dei requisiti di accessibilità definiti dal W3C (ad esempio, si veda la lista di tool presente in <http://www.w3.org/WAI/ER/existingtools.html>). Scopo di questo progetto è lo sviluppo di un *semplice* tool di validazione dell'accessibilità basato su trasformazioni XSLT. In questo progetto si richiede di procedere come segue:

a) individuare alcuni requisiti di accessibilità tra quelli descritti in “*Web Content Accessibility Guidelines 1.0*” che possano essere verificati in modo automatico; siete

liberi di imporre vincoli ancora più stretti di quelli richiesti dalle specifiche W3C. Ad esempio, il requisito 13.5 recita: “*Provide navigation bars to highlight and give access to the navigation mechanism*”; se volete validare questo requisito, potreste decidere di rendere obbligatoria la presenza all'interno della pagina di una barra di navigazione che sia tassativamente contenuta all'interno di un blocco `<map> .. </map>` con `id="navbar"`.

b) scrivere quindi una trasformazione XSLT che accetti in input una pagina XHTML, e produca in output la lista degli eventuali elementi della pagina in input che violino le regole di accessibilità considerate. Siete liberi di produrre l'output nel modo che vi sembra più opportuno. Un esempio potrebbe essere quello di generare in output la stessa pagina data in input, con delle annotazioni chiaramente visibili (ad esempio usando colori di sfondo differenti) sugli elementi che violino le regole da voi considerate. Oppure potreste decidere di produrre in output la lista delle violazioni dei requisiti di accessibilità (se ce ne sono).

Nella documentazione allegata al progetto è necessario descrivere tutte le scelte fatte al punto a), fornire lo stylesheet XSLT di cui al punto b), e fornire alcuni esempi di pagine XHTML che soddisfano o non soddisfano i requisiti definiti in a).

2. Lo scopo di questo progetto è di produrre un tool minimale per l'analisi dei percorsi degli utenti all'interno di un sito WEB. Il tool in questione riceve in input un file di log di Apache, e produce in uscita un file contenente le istruzioni per produrre un grafo che rappresenti sinteticamente informazioni sulle visite alle pagine di cui al file di log. L'output prodotto dal programma deve contenere istruzioni utilizzabili dal programma `dot` del toolkit `GraphViz` (<http://www.research.att.com/sw/tools/graphviz/>) per costruire graficamente un grafo orientato in cui i nodi rappresentano le pagine WEB elencate nel log di Apache, ed esiste un arco che va dal nodo X al nodo Y se esiste una sessione di visita all'interno del log in cui lo stesso utente (identificato tramite il suo indirizzo IP) visita la pagina Y immediatamente dopo aver visitato la pagina X; lo spessore dell'arco tra due nodi X e Y è proporzionale alla probabilità con cui un visitatore acceda alla pagina Y immediatamente dopo aver visitato la pagina X. In più, i vari nodi del grafo vanno evidenziati in modo opportuno (ad esempio, con diverse gradazioni di un colore a vostra scelta, oppure con diversi spessori del bordo) in base alla probabilità che quel nodo sia il nodo di ingresso o di uscita di una sessione di navigazione sul sito. L'analizzatore di log deve essere sviluppato in C/C++ oppure in Java (a vostra scelta). La documentazione deve includere una parte in cui descrivete in dettaglio l'algoritmo con cui determinate i percorsi, algoritmo che siete liberi di sviluppare nel modo da voi ritenuto più corretto (purché le vostre scelte siano descritte e ove possibile motivate). Allegare al progetto, oltre alla documentazione e ai sorgenti, un esempio di log su cui è possibile testare il vostro programma.

Tesine:

1. Descrivere il funzionamento del protocollo R-HTTP (Reliable HTTP). Far riferimento alle specifiche sul sito <http://www-128.ibm.com/developerworks/library/ws-httpspec/> e all'introduzione presente in <http://www-128.ibm.com/developerworks/webservices/library/ws-phhttp/>

2. Descrivere il funzionamento di un meccanismo a vostra scelta per la comunicazione anonima tramite posta elettronica; far riferimento alla bibliografia presente in http://freehaven.net/anonbib/topic.html#Anonymous_20communication