

Gestione dei dati – appello del 13/4/2012

COGNOME:
NOME:
MATRICOLA:

Autorizzo la pubblicazione del mio voto di questo esame sul sito web http://www.dis.uniroma1.it/~rosati/gd , secondo quanto prevede il decreto legislativo 196/2003 (codice in materia di protezione dei dati personali) che dichiaro di conoscere. In fede,

Esercizio 1 Dato il seguente schedule $S = r_3(D) r_5(A) r_5(B) r_2(C) w_1(B) r_4(B) r_1(C) w_4(C) r_4(D) r_1(A) w_2(D) r_2(E) w_3(E) r_3(A)$

1. dire se S è view-serializzabile, motivando la risposta;
2. dire se S è eseguibile da uno scheduler che segue il protocollo 2PL con lock esclusivi e condivisi. In caso positivo, completare lo schedule S con le opportune istruzioni di lock e unlock. In caso negativo, motivare la risposta;
3. dire se S è eseguibile da uno scheduler che segue il protocollo 2PL stretto. In caso positivo, completare lo schedule S con le opportune istruzioni di lock e unlock. In caso negativo, motivare la risposta;
4. dire se è possibile inserire in S le istruzioni di commit delle transazioni T_1, \dots, T_5 in modo tale che lo schedule risultante sia ACR e, in caso positivo, scrivere tale schedule.

Esercizio 2 Si consideri una base di dati contenente la relazione **Impiegato** con attributi **matricola**, **cognome**, **nome**, **indirizzo**, **annoDiAssunzione**, **stipendio**, **annoDiNascita**. La relazione **Impiegato** contiene 10^3 record, e ogni record di tale relazione ha dimensione $N/10$, dove N è la dimensione di una pagina di memoria. Inoltre **matricola** è chiave per la relazione **Impiegato**. La relazione **Impiegato** è soggetta ad aggiornamenti.

1. Si considerino query del tipo: “selezionare la matricola di tutti gli impiegati assunti in un certo anno a e il cui stipendio è compreso tra s_1 e s_2 ”. Scegliere l’organizzazione fisica per la relazione **Impiegato** che minimizza il costo dell’esecuzione di tali query;
2. assumendo un tempo medio di accesso ad una pagina in memoria di massa di 5 millisecondi, e trascurando i tempi di elaborazione in memoria centrale, calcolare il tempo (espresso in millisecondi) necessario all’esecuzione delle query di cui al punto 1 in base all’organizzazione fisica per la relazione **Impiegato** scelta al punto 1;
3. si considerino ora query del tipo: “selezionare lo stipendio e l’anno di nascita di tutti gli impiegati il cui cognome è c , che hanno stipendio compreso tra s_1 e s_2 e il cui anno di assunzione è compreso tra a_1 e a_2 ”. Scegliere l’organizzazione fisica per la relazione **Impiegato** che minimizza il costo dell’esecuzione di tali query;
4. assumendo un tempo medio di accesso ad una pagina in memoria di massa di 5 millisecondi, e trascurando i tempi di elaborazione in memoria centrale, calcolare il tempo (espresso in millisecondi) necessario all’esecuzione delle query di cui al punto 3 in base all’organizzazione fisica per la relazione **Impiegato** scelta al punto 3.

Esercizio 3 Data la seguente DTD:

```
<!DOCTYPE b [  
  <!ELEMENT b (a+)>  
  <!ELEMENT a (p,(t|(q,a)|(y,a)),(y,t)*)>  
  <!ELEMENT p (s,t*(q|y))>  
  <!ELEMENT q (t|y|p|a)>  
  <!ELEMENT t (#PCDATA)>  
  <!ELEMENT y (y|a|s)*>  
  <!ELEMENT s (#PCDATA)>  
  <!ATTLIST a val CDATA #REQUIRED>  
  <!ATTLIST p val CDATA #REQUIRED>  
  <!ATTLIST s attr CDATA #REQUIRED>  

```

1. Scrivere lo schema relazionale ottenuto applicando l’algoritmo schema-driven di XML shredding alla DTD precedente.
2. Scrivere un documento XML valido rispetto alla DTD precedente e scrivere l’istanza della base di dati ottenuta traducendo tale documento nello schema relazionale ottenuto al punto 1.

Esercizio 4 Si consideri una base di dati contenente la relazione **Esame** con attributi **codStudente**, **codCorso**, **voto**, **data** e la relazione **Corso** con attributi **codCorso**, **nomeCorso**, **codDocente**, **numCrediti**, **descrizione**. La relazione **Esame** contiene 40.000 record, e ogni record di tale relazione ha dimensione $N/20$, dove N la dimensione di una pagina di memoria. Su tale relazione è dichiarato un indice B⁺-tree clusterizzato (che usa l’alternativa 1) con chiave di ricerca **data**.

La relazione **Corso** contiene 800 record, e ogni record di tale relazione ha dimensione $N/5$. Inoltre, **codCorso** è chiave per la relazione **Corso**, e su tale relazione è dichiarato un indice B⁺-tree clusterizzato (che usa l’alternativa 1) con chiave di ricerca **codCorso**.

Si consideri la seguente query Q:

```
select E.codStudente, C.nomeCorso from Esame E, Corso C  
where E.codCorso = C.codCorso and C.numCrediti = "6"
```

Si assuma che: (i) il sistema possa eseguire i join sia mediante l’algoritmo Index Nested Loop che mediante l’algoritmo Sort Merge Join; (ii) il buffer abbia a disposizione 12 slot liberi per eseguire la query; (iii) la relazione **Corso** abbia 100 record con valore dell’attributo **numCrediti** pari a 6. Si richiede di scrivere il query plan per tale query che minimizza il numero di accessi alla memoria di massa, e di calcolare il costo dell’esecuzione di tale query plan, espresso in numero di accessi alla memoria di massa.