

Gestione dei dati – appello del 25/7/2011

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 $S = w_4(F) r_1(E) r_2(G) w_3(H) w_3(G) r_5(F) w_5(D) w_1(H) r_2(E) r_1(D) r_4(G)$

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 è possibile inserire in S le istruzioni di commit delle transazioni T_1, \dots, T_5 in modo tale che lo schedule risultante sia recuperabile e, in caso positivo, scrivere tale schedule;
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 stretto e, in caso positivo, scrivere tale schedule.

Esercizio 2 Data la seguente DTD:

```
<!DOCTYPE a [  
  <!ELEMENT a (b+, (c,g)+, (d|e)*, f) >  
  <!ELEMENT b (#PCDATA)>  
  <!ELEMENT c (b*, d, g) >  
  <!ELEMENT d (b|c|d)*>  
  <!ELEMENT e (b,d)>  
  <!ELEMENT f (#PCDATA)>  
  <!ELEMENT g ((d|g)*, c*)>  
  <!ATTLIST a v CDATA #REQUIRED>  
  <!ATTLIST c w 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 3 Sia data la seguente query:

```
SELECT T2.B, T2.C  
FROM T1, T2  
WHERE T2.D="Roma" AND T1.C = T2.C
```

e si assuma che sulla relazione $T1$ sia dichiarato un indice hash non clusterizzato con chiave di ricerca C , che sulla relazione $T2$ sia dichiarato un indice hash non clusterizzato con chiave di ricerca D , e che il DBMS possa eseguire i join sia mediante l'algoritmo block nested loop che mediante l'algoritmo sort merge join. Si assuma inoltre che $T1$ sia contenuta in 150 pagine, $T2$ sia contenuta in 80 pagine, che i record di $T2$ con $D="Roma"$ siano contenuti in 3 pagine e che il DBMS abbia a disposizione 10 pagine per eseguire i join.

Si richiede di: (1) scrivere il migliore query plan per tale query; (2) calcolare il costo dell'esecuzione del query plan, espresso come numero di trasferimenti di pagine da memoria di massa.

Esercizio 4 Si consideri una base di dati contenente la relazione **Studente** con attributi **cognome**, **nome**, **codiceFiscale**, **matricola**, **annoDiNascita**, **comune**, **indirizzo**. La relazione **Studente** contiene 10^5 record, e ogni record di tale relazione ha dimensione $N/250$, dove N è la dimensione di una pagina di memoria. Inoltre **matricola** è chiave per la relazione **Studente**. La relazione **Studente** è soggetta a frequenti aggiornamenti.

1. Si considerino query del tipo: "selezionare matricola e indirizzo di tutti gli studenti aventi un certo cognome n e un anno di nascita compreso tra a_1 e a_2 ". Scegliere l'organizzazione fisica per la relazione **Studente** 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 **Studente** scelta al punto 1;
3. si considerino ora query del tipo: "selezionare cognome, nome e indirizzo di tutti gli studenti aventi una certa matricola m e nati dopo un certo anno a_1 e prima di un certo anno a_2 ". Scegliere l'organizzazione fisica per la relazione **Studente** 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 **Studente** scelta al punto 3.