

# Gestione dei dati – appello del 23/6/2010

|                  |
|------------------|
| COGNOME: .....   |
| NOME: .....      |
| MATRICOLA: ..... |

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

**Esercizio 1** Si consideri una base di dati contenente la relazione **Residente** con attributi **cognome**, **nome**, **dataNascita**, **codiceFiscale**, **nomeComune**, **nomeProvincia** e la relazione **Comune** con attributi **nomeComune**, **numeroAbitanti**, **nomeProvincia**, **superficie**.

Si assuma che la relazione **Residente** contenga  $10^6$  record, e che ogni record di tale relazione abbia dimensione  $N/200$ , dove  $N$  la dimensione di una pagina di memoria. Si assuma inoltre che la relazione **Comune** contenga  $10^4$  record, e che ogni record di tale relazione abbia dimensione  $N/200$ . Si supponga inoltre che **(nomeComune,nomeProvincia)** sia chiave per la relazione **Comune**.

Si consideri ora la query:

```
select R.cognome, R.nome, C.superficie
from Residente R, Comune C
where R.nomeComune = C.nomeComune and R.nomeProvincia = C.nomeProvincia
```

- scegliere l'organizzazione fisica per le relazioni **Residente** e **Comune** che minimizza il tempo di esecuzione della precedente query nell'ipotesi che il sistema esegua l'algoritmo index nested loop, e calcolare tale tempo;
- scegliere l'organizzazione fisica per le relazioni **Residente** e **Comune** che minimizza il tempo di esecuzione della precedente query nell'ipotesi che il sistema esegua l'algoritmo block nested loop, avendo a disposizione 12 slot (pagine) per valutare la query, e calcolare tale tempo.

**Esercizio 2** Dato il seguente schedule  $S$ :

$$S = r_1(A) r_3(C) r_1(B) w_2(D) w_0(B) w_0(C) r_4(A) r_2(D) r_3(D) w_1(D) r_2(E) c_1 w_3(E) c_2 c_3 w_4(B) w_4(E) c_4$$

1. dire se  $S$  è conflict-serializzabile, motivando la risposta;
2. dire se  $S$  è eseguibile da uno scheduler che segue il protocollo 2PL stretto 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 ACR, motivando la risposta;
4. dire se  $S$  è stretto, motivando la risposta.

**Esercizio 3** Data la seguente DTD:

```
<!DOCTYPE root [
  <!ELEMENT root (impiegato*) >
  <!ELEMENT impiegato (matricola, cognome, nome+, sede) >
  <!ELEMENT matricola (#PCDATA)>
  <!ELEMENT cognome (#PCDATA)>
  <!ELEMENT nome (#PCDATA)>
  <!ELEMENT sede (nome, responsabile*)>
  <!ELEMENT responsabile (nome|impiegato)* >
  <!ATTLIST impiegato dataNascita CDATA #REQUIRED
    reparto CDATA #IMPLIED>
]>
```

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 **Studente** con attributi **nome**, **cognome**, **matricola**, **annoDiNascita**, **comuneDiResidenza**, **provincia**. La relazione **Studente** non è soggetta a frequenti aggiornamenti. Inoltre **matricola** è chiave della relazione. Su tale relazione esiste un indice clusterizzato con chiave di ricerca **matricola**.

1. Si considerino ora le query del seguente tipo: dato un anno  $a_1$  ed un anno  $a_2$ , restituire cognome, nome e matricola di tutti gli studenti nati in un anno maggiore o uguale ad  $a_1$  e minore di  $a_2$ . Dire quale organizzazione di file è la più indicata per la relazione **Studente**, motivando la risposta (l'organizzazione di file scelta deve comunque mantenere l'indice sopra descritto);
2. sulla base dell'organizzazione di file scelta al punto 1 precedente, illustrare la funzione che caratterizza il costo del calcolo della risposta a una query del tipo descritto precedentemente rispetto al numero di pagine necessarie per memorizzare la relazione;
3. si supponga ora che le query di gran lunga più frequenti per la relazione **Studente** siano del seguente tipo: dato un anno  $a$ , restituire cognome, nome e matricola di tutti gli studenti nati nell'anno  $a$ . Sotto tale ipotesi, dire quale organizzazione di file è la più indicata per la relazione **Studente**, motivando la risposta;
4. sulla base dell'organizzazione di file scelta al punto 3 precedente, illustrare la funzione che caratterizza il costo del calcolo della risposta a una query del tipo descritto precedentemente rispetto al numero di pagine necessarie per memorizzare la relazione.