Data management – 5/6/2014

LAST NAME: _____

FIRST NAME: _____

ID (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 Given the following schedule S:

$$S = w_3(E)\, r_2(D)\, w_2(C)\, r_5(B)\, r_3(F)\, r_1(D)\, r_3(D)\, w_5(F)\, r_2(E)\, w_2(B)\, r_5(D)\, w_4(C)\, w_2(D)\, r_4(F)\, r_4(D)$$

1. tell whether S is conflict-serializable, explaining your answer;
2. tell whether S is view-serializable, explaining your answer;
3. tell whether S can be executed under the 2PL protocol (with both exclusive locks and shared locks). If this is possible, complete the schedule S with the proper lock/unlock instructions. Otherwise, explain why it is not possible;
4. tell whether it is possible to insert the commit instructions of transactions $T_1, \ldots, T_5$ in S in such a way that the resulting schedule is strict. If this is possible, then write such a schedule. Otherwise, explain why it is not possible;
5. tell whether it is possible to insert the commit instructions of transactions $T_1, \ldots, T_5$ in S in such a way that the resulting schedule is recuperable. If this is possible, then write such a schedule. Otherwise, explain why it is not possible.

Esercizio 2 Consider a database containing a relation `Employee` with attributes `employeeID, lastName, firstName, birthDate, city, salary`. Relation `Employee` contains 20.000 records, and the size of each record is N/40, where N is the size of a memory page. Moreover, the attribute `employeeID` is a key for the relation. Relation `Employee` is never updated.

1. Consider queries of the following form: "Select the salary of employees having first name f and living in city c". Choose the physical organization for relation `Employee` that minimizes the execution cost of such queries;
2. assuming an average access time to a mass memory page of 5 milliseconds, and ignoring the time needed for processing data in central memory, compute the time (in milliseconds) needed to execute the queries specified at point 1 according to the physical organization chosen at the above point 1;
3. now consider queries of the following form: "Select the ID of employees having last name l and whose salary is not less than s_1' and not greater than s_2'. Choose the physical organization for relation `Employee` that minimizes the execution cost of such queries;
4. assuming an average access time to a mass memory page of 5 milliseconds, and ignoring the time needed for processing data in central memory, compute the time (in milliseconds) needed to execute the queries specified at point 3 according to the physical organization chosen at the above point 3.

Esercizio 3 Consider a database containing relation `Movie`, whose attributes are `title, year, directorID, genre, duration, country`, and relation `Director`, whose attributes are `directorID, name, country, birthDate`. Relation `Movie` contains 40.000 records, and the size of every such record is N/50, where N is the size of a memory page. Relation `Director` contains 5000 records, and the size of every such record is N/25. Moreover, the attribute `directorID` is a key for relation `Director`.

Now consider the following query Q:

```
SELECT F.title, R.name
FROM Movie F, Director R
WHERE F.directorID = R.directorID
```

and assume that the buffer has 32 available slots for the execution of the query.

1. Assuming that joins are executed through the Block Nested Loop algorithm, choose the physical organization of the relations `Movie` and `Director` such that the number of accesses to mass memory pages during the execution of the above query is minimum;
2. assuming that joins are executed through the Block Nested Loop algorithm, and ignoring the time needed for processing data in central memory, compute the time (in milliseconds) needed to execute the above query according to the physical organization chosen at the above point 1;
3. assuming that joins are executed through the Index Nested Loop algorithm, choose the physical organization of the relations `Movie` and `Director` such that the number of accesses to mass memory pages during the execution of the above query is minimum;
4. assuming that joins are executed through the Index Nested Loop algorithm, and ignoring the time needed for processing data in central memory, compute the time (in milliseconds) needed to execute the above query according to the physical organization chosen at the above point 3.