Data Management – exam of 04/03/2011

LAST NAME: _____

FIRST NAME: _____

STUDENT CODE: _____

I allow the publication of the grade I will get in this exam on the web page `http://www.dis.uniroma1.it/~lenzerini`, according to the "decreto legislativo 196/2003" (regarding the rules for the privacy of personal data) which I hereby confirm to know. Sincerely,

(Signature) _____

Problem 1 Consider the schedule $S = w_4(W)\, r_1(Z)\, r_2(X)\, w_3(Y)\, w_3(X)\, r_5(W)\, w_1(X)\, r_2(Z)\, r_4(T)$

1. Tell whether S is conflict-serializable or not, and explain the answer.
2. Tell whether S is accepted by the 2PL scheduler with exclusive and shared locks. If the answer is yes, then show the schedule obtained from S by adding suitable lock and unlock commands coherently with the 2PL protocol. If the answer is no, then explain the answer.
3. Tell whether we can insert in S the commit commands of transactions $T_1, \ldots, T_5$ in such a way that the resulting schedule is ACR (avoids cascading rollback), and explain the answer.
4. Tell whether we can insert in S the commit commands of transactions $T_1, \ldots, T_5$ in such a way that the resulting schedule is recoverable, and explain the answer.

Problem 2 Prove or disprove the following statements (illustrating the proof in detail):

1. If S is a conflict-serializable schedule on transactions $\{T_1, \ldots, T_n\}$ constituted only by write actions, then we can insert the commit actions for the transactions $T_1, \ldots, T_n$ in S in such a way that the resulting schedule is strict.
2. If S is a strict schedule on transactions $\{T_1, \ldots, T_n\}$ constituted only by write and commit actions, then S is conflict-serializable.

Problem 3 Consider a data base with relation `Teaches(prof,course)`, and relation `Professor(prof,age)`. Relation `Teaches` has 1.000.000 tuples, with 10 tuples fitting in one page, and is stored as a Heap file. Relation `Professor` has `prof` as key, has 100.000 tuples, with 10 tuples fitting in one page, and has an associated clustered B^+-tree index on attribute `age` using alternative 1. Tell which is the cost (in milliseconds) of the natural join between `Teaches` and `Professor`, assuming that

- the buffer has 10 free slots for computing the join;
- the system uses the "sort merge join" algorithm for computing the join;
- the average time for accessing one page is 5 milliseconds;
- the time for main memory operations is ignored,

and motivating the answer in detail.

Problem 4 There are two main methods for representing a Heap File in secondary storage, namely "through lists", and "through directory". Illustrate and compare the two methods.

Problem 5 Consider the relation `Product(code,price,size)`, where `code` is a key, storing information about a set of products to be sold by a company. We know that the relation `Product` has 1.000.000 tuples, that every page has room for 100 tuples of the relation, that each value of all attributes of the relation requires M bytes, that M/2 bytes are needed to store page ids, that M/2 bytes are needed to store slot ids, and that, in the average, for each price, we have 100 products of that price. The most important queries on `Product` are the following:

1. Given a product code c, find the price and the size of the product whose code is c (run 5 times per day).
2. Find the code and the size of all products whose price is in a given range (run 3 times per day).

Which is the file organization you would choose for the relation? Assuming for the relation the chosen file organization, which is the daily cost of answering the two queries in terms of page accesses? Motivate both answers in detail.