Data Management – exam of 15/06/2018

Problem 1 Consider the following schedule

$$S = r_1(C, v_{11}),\ v_{21} := 20,\ w_2(A, v_{21}),\ w_1(A, v_{11}),\ v_{12} := 30,\ w_1(B, v_{12}),\ r_2(B, v_{22}),\ w_2(C, v_{22}).$$

where the notation $r_i(X, v)$ is used to denote the action by transaction i that reads the element X of the database and stores the value in the local variable v, and the notation $w_i(X, v)$ is used to denote the action by transaction i that writes the value of the local variable v into the element X of the database.

1. Tell whether S is serializable or not. If the answer is yes, then motivate the answer in detail. If the answer is no, then explain the answer, and tell whether there is a single action we can add to the schedule in order to make it serializable, illustrating such action, and explaining why its addition makes the schedule serializable.
2. Tell whether S is recoverable, explaining the answer.
3. Tell whether S is strict, explaining the answer.

Problem 2 By referring to the following schedule S:

$$w_1(x)\, r_2(y)\, w_1(y)\, c_1\, r_3(x)\, w_3(x)\, c_3\, w_2(t)\, c_2\, w_5(y)\, w_4(z)\, w_5(z)\, r_5(v)\, c_5\, w_4(y)\, c_4$$

you are asked to:

- Illustrate the various steps carried out by the timestamp-based scheduler when analyzing S, assuming that, initially, rts(α)=wts(α)=wts-c(α)=cb(α)=`true` for each element α of the database, and assuming that the timestamp of each transaction T_i is i.
- Tell whether S is conflict serializable.
- Tell whether S is in 2PL with shared and exclusive locks.

Problem 3 Describe in the most detailed way possible the parallel algorithm for computing the natural join between $R(X, Y)$ and $S(Y, Z)$, and tell which is the cost of the algorithm, both in terms of number of page accesses, and in terms of elapsed time.

Problem 4 Consider the relations `R(A,B,C)` and `Q(D,E,F,G,H,L)`, where `D,E` form the key of the relation `Q`, and consider the join of the two relations on the condition `C` = `D`. We know that `R` has 9.000 pages, every page of `R` contains 90 tuples, `Q` has 10.000 pages, every page of `Q` contains 45 tuples, the buffer has 97 pages free, and there is a non-clustering tree-based index on `Q` with search key `D,E`. We also know that for every value of `D` in `Q`, there are an average of 5 different tuples in `Q` with that value in `D`. On the basis of the above scenario, answer the following questions:

1. Is the “block nested loop” algorithm applicable for computing the join? If yes, which is the cost of the join in this case? Motivate the answer.
2. Is the “two-pass sort-merge join” algorithm applicable for computing the join? If yes, which is the cost of the join in this case? Motivate the answer.
3. Is the “index nested loop” algorithm applicable for computing the join? If yes, which is the cost of the join in this case? Motivate the answer.

Problem 5 Consider the relations `R(A,B,C)` and `S(A,B,C)`, where `A` is the primary key in both relations, and there is a hash-based index on `S(A,B,C)` with search key `A`. Describe in detail an index-based algorithm for computing the difference between `R(A,B,C)` and `S(A,B,C)`.