Data Management – exam of 31/01/2013

Problem 1 An *extended schedule* is a schedule expressed in an extended notation, that allows the schedule to contain assignment statements involving local variables, and enriched read and write statements of the form $read_i(B, x)$ and $write_i(B, x)$, where the statement $read_i(B, x)$ means that transaction i reads the database element B and stores its value in the local variable x, and the statement $write_i(B, x)$ means that transaction i writes the value of the local variable x in the database element B. Consider the following extended schedule S:

$$v_1 := 0; read_1(A, w_1);\ w_1 := v_1 * w_1;\ read_2(A, w_2);\ w_2 := w_2 - w_2;\ write_2(A, w_2);\ write_1(A, w_1);.$$

1.1 Tell whether S is serializable or not. Explain the answer in detail.
1.2 Tell whether S is view-serializable or not. Explain the answer in detail.

Problem 2 A *1-write schedule* is a schedule containing exactly one write action. Give the definition of view-serializable schedule, and, using only such definition, prove or disprove that every 1-write schedule is view-serializable.

Problem 3 Consider the following schedule

$$S = r_1(A)\, r_3(C)\, w_3(B)\, r_2(A)\, w_1(B)\, r_2(C)\, w_3(C)\, r_3(A)\, w_2(D).$$

3.1 Tell whether S is a 2PL schedule with exclusive and shared locks, explaining the answer in detail.
3.2 Tell whether S is conflict-serializable or not. If the answer is yes, then show a serial schedule that is conflict-equivalent to S. If the answer is no, then motivate the answer in detail.
3.2 Tell whether S is strict or not, and whether S is ACR or not, explaining the answer in detail.

Problem 4 Suppose that page P in our Data Base Management System is a page with fixed-length records containing 100 slots, and suppose we ask for the deletion of the record with $rid{=}\langle P, 10\rangle$. Illustrate the various actions that the system performs in the two cases of packed and unpacked organization for P, respectively.

Problem 5 Consider the relation `TENNISCHOOL(code,city,cost,numstud)` that stores 200.000 tuples, where each tuple $\langle d, t, c, n\rangle$ means that the tennis school d is in city t, has a monthly cost of c, and has n students. We assume that (i) in the average, for every value v of `cost`, there are 20 schools whose monthly cost is v, (ii) in the average, for every value w of `numstud`, there are 10 schools whose number of student is w, (iii) every page in secondary storage has space for 10 records of the relation `TENNISCHOOL`, (iv) every attribute and every record id occupies the same space in the page, and (v) the following are the most important queries on `TENNISCHOOL` (where Query 1 is even more important than Query 2):

Query 1	**Query 2**
select `code` from `TENNISCHOOL` where `cost` $\geq \alpha$ and `cost` $\leq \alpha + 10$	select `code`, `city` from `TENNISCHOOL` where `numstud` $\geq \beta$ and `numstud` $\leq \beta + 2$

Tell which is the method for representing the relation `TENNISCHOOL` you would choose in order to optimize the computation of the above queries, explaining in detail your answer. Also, tell how many pages are accessed during the execution of Query 1, and how many pages are accessed during the execution of Query 2.