Data Management – AA 2017/18 – exam of 12/01/2018

Problem 1
We have to compute the bag difference between the two tables `R(A,B,C)` and `S(A,B,C)`, where each table is stored in a file sorted on `<A,B,C>`. We know that `R` occupies 4.000 pages, and `S` occupies 5.000. Describe in detail the algorithm you would use to perform the operation, knowing that you have 100 buffer frames available. Also, tell which is the cost of such algorithm in terms of number of page accesses, explaining the answer in detail.

Problem 2
Let `R(A,B,C)` and `S(D,E,F,G)` be two relations, where `R` is stored in 7.500 pages, and the 300.000 tuples of `S` are stored in 3.000 pages. We know that `S` has a dense, clustering B^+ index with search key `E` using alternative 2, every value or pointer occupies the same space, and our system has 501 free buffer frames available. Consider the query

```
select A,B,F,G
from S, R
where E >= 50 and C = D
```

2.1 Describe the logical query plan associated to the query code.
2.2 Describe the selected logical query plan, explaining why such logical plan has been selected.
2.3 Describe the selected physical query plan, explaining why such physical plan has been selected.
2.4 Tell which is the cost (in terms of number of page accesses) of executing the query according to the selected physical query plan.

Problem 3
Consider the relation `TRAVEL(code,agency,country,date,cost)`, storing for each travel its code (primary key), the agency that organized it, the country visited during the travel, and the starting date of the travel. The relation has 640.000 tuples stored in a heap file, where each page contains 80 tuples. Consider the query Q that, for each agency a, computes the average cost of the travels organized by a, and assume that we have a good hash function on `agency` that distributes the tuples of `TRAVEL` uniformly into 100 buckets. You are asked to describe the algorithm you would use for computing Q, and tell which is the cost of executing the algorithm in terms of number of page accesses, in the following two scenarios:

3.1 under the assumption that we have only the processor where `TRAVEL` is stored, with 101 free buffer frames available;
3.2 under the assumption that we have 100 processors besides the one where `TRAVEL` is stored, each one with 90 free buffer frames available.

Problem 4
Consider the following schedule

$$S = r_1(x)\, r_2(y)\, w_2(x)\, r_3(v)\, w_1(z)\, w_4(v)\, r_1(y)\, w_4(z)\, w_2(z)\, r_3(z).$$

4.1 Tell whether S is a 2PL schedule or not, explaining the answer in detail.
4.2 Construct the precedence graph associated to S, and tell whether S is conflict-serializable or not, explaining the answer in detail.
4.3 Describe the behavior of the timestamp-based scheduler when processing the schedule S, assuming that, initially, rts(α)=wts(α)=wts-c(α)=0, and cb(α)=`true` for each element α of the database, and assuming that the subscript of each action denotes the timestamp of the transaction executing such action.
4.4 Tell whether S is in the class of ACR (Avoiding Cascading Rollback) schedules or not, explaining the answer in detail.
4.5 Tell whether S is in the class of strict schedules or not, explaining the answer in detail.

Problem 5
A schedule is called "conflict-restricted" if it has no conflict of type `write-write`, and no conflict of type `read-write`. Proof or disprove the following claim: every "conflict-restricted" schedule that is in the class ACR (Avoiding Cascading Rollback) is conflict-serializable.