Final take-home exam for CSE6421 Winter 2012

No collaboration on exam questions is permitted

The exam is due (by email or hardcopy) on Dec 17 at 9AM.

Questions 1-12 are 3 points each; questions 13-14 are 2 points each.

1. Suppose 10 tuples of R can fit in one block, R has 100,000 tuples, and there are among these tuples 20,000 distinct tuples. We wish to eliminate duplicates in one pass. That is, we read blocks of R into a single buffer, one at a time, use B buffers (block-sized chunks of main memory) to hold intermediate results, and write the answer somewhere (it is not important where the output goes). What is the smallest possible value of B?

2. Assume we have the following dependencies between views: $d \leq b, d \leq c, c \leq a, b \leq a$. The views a, b, c, d (if materialized) would have the sizes 10, 5, 10, and 1 respectively. Specify in which order the views will be chosen for materialization by the greedy algorithm and with what benefits. You can assume that the top view is already materialized and that there is infinite storage.

3. Why is *Manhattan distance* (rather than another metric such as LRU or MRU) used in the replacement strategy of semantic query caching.

4. A B+ tree on a certain data file has three levels of nodes with an order equal 25 (that is, for the maximum of 50 search keys in a node). We assume it is maintained as described in class or the text; in particular, subminimum blocks due to deletion are merged and not allowed to exist. What is the minimum number of records in the data file?

5. Does the COUNT bug still exist if COUNT is replaced with MAX? Explain briefly why or why not.

6. For which of the join methods discussed in class will the empty join technique (recall that the technique consists in reducing a range of one (or more) of the attribute of a query) work? For which it will not work? Explain briefly why or why not.

7. Assume that selectivity for predicate p is $s_p = 0.5$ and the selectivity for predicate q is $s_q = 0.6$, yet the selectivity for a query with both p and q is $s_{p \wedge q} = 0.5$. What does it say about the predicates p and q?

8. Exercise 11.6 from the textbook. Assume the directory is one page.

9. Give an example of an SQL operator that *is* always commutable with sampling. Justify your choice with a short argument.

10. In the execution of a hash join of **R** and **S**, is a record of **R** ever compared with a record of **S** *and* they do *not* match (on the join attribute(s))? If not, briefly explain how the hash join method avoids this. If so, give an example of how this can occur.

11. Suppose relation R occupies N and relation S occupies M blocks. We assume that N $\leq$ M. Initially both relations are on disk, and we want to perform a nested-loop join in which we read each block of R and S only once (and do no other disk I/O other than writing the result). What is the minimum number of main-memory buffers (each the size of a block) that we need?

12. Consider an extensible hash table where 200 buckets are actually allocated at th is point. What is the size (in the number of entries) of the smallest possible directory in this case?

13. Consider table R with attributes A and B, table S with attributes B and C, and table T with attributes C and D. All joins below are natural joins; that is, the join columns are those columns named the same between the two tables.

    I $\pi_B(R \bowtie (S \bowtie T))$

    II $((\pi_B(R \bowtie S)) \bowtie T)$

    III $((\pi_B(R) \bowtie S) \bowtie T)$

    Which of the above relational algebra expressions necessarily evaluate to the same result?

    (A) All three evaluate to the same result.

    (B) They each evaluate to different results.

    (C) I and II

    (D) I and III

    (E) II and III

    (F) Not enough information is provided to determine this.

14. We wish to join the relations R(a, b), S(b, c), and T (c, d) under the following assumptions:

    1. There are B buffers available to hold blocks of data from these three relations.
    2. The relations occupy n_R, n_S , and n_T blocks, respectively.
    3. R is stored sorted by b; the other relations are unsorted.
    4. The particular strategy we shall use to perform the join is:

    i) Perform the first phase of two-phase multiway merge sort on S. That is, as many times as necessary, load all buffers from S, sort the tuples on b, and write out the sorted sublist (assume no further optimizations here).

    ii) Load T entirely into main memory, using as many buffers as needed.

    iii) Merge (and join when appropriate) R and the sorted sublists of S, and compare each of the resulting tuples with the tuples of T . Any join tuple in the result is stored in an output buffer, not counted among the B buffers available for this process.

    Which of the following inequalities is the closest approximation to the condition under which this sequence of steps can be carried out as described?

(a) $B^2 \geq n_S + Bn_T$

(b) $B^2 \geq n_T + Bn_S$

(c) $B^2 \geq n_T + n_S$

(d) $B^2 \geq n_R + Bn_T + B^2 n_S$

(e) $B^2 \geq n_R + n_T + Bn_S$