Name: C.S.

Department of Computer Science
California State University, Los Angeles

December 5, 2004
CS522 Homework Assignment 1

---

1. Exercise 15.2.4 (c) and (d)

   (a)
   - Read R into some in-memory data structure that supports efficient *delete*, e.g. a hash table or a balanced binary search tree.
   - Read S into memory one page at a time.
   - For each tuple s in S, remove from R all tuples that agree with s in the common attributes.
   - Output the remaining tuples of R.

   (b)
   - Read S into some in-memory data structure that supports efficient *search*, e.g. a hash table or a balanced binary search tree.
   - Read R into memory one page at a time.
   - For each tuple r in R, output r if r does *not* agree with any tuple in S in the common attributes.

2. Exercise 15.4.10

   Consider TPMMS. Let R be the relation to be sorted, B be the number of disk pages for R, and M be the number of memory pages. TPMMS consists of the following steps:

   (a) Divide R into k sublists, where

   $$k = \lceil \frac{B}{M} \rceil <= M - 1$$

   or in other words, each sublist contains the max number of tuples that can fit into memory, and the total number of sublists is less than M because we need one memory page as the output buffer.

   (b) Read in each sublist and perform in-memory sort, then write out the sorted sublist.

   (c) Read in one page from each sublist, merge the sublists, and output the results.

   Note that the I/O complexity of TPMMS is 3B since all data pages have to be read in, write out, then read in again.

   Now suppose the size of the last sublist is X. If we keep the last sublist in memory, we save $2X$ I/O, so the problem becomes how we can maximize X. Note that the best we can do is:

   $$X + k - 1 + 1 = M$$

or in other words, X can be as large as $M - k$, because we need $k - 1$ pages to read in one page from each of the other sublists, and one page for output buffer. Also note that we have

$$k = \lceil \frac{B}{X} \rceil$$

Combine the two equations, we have a quadratic equation:

$$X^2 - MX + B = 0$$

From the quadratic formula,

$$X = \frac{M \pm \sqrt{M^2 - 4B}}{2}$$

So the I/O saving is $2X = M + \sqrt{M^2 - 4B}$.

3. Exercise 16.2.8

   Intuitively, we cannot swap MIN and SUM because MIN has the effect of eliminating duplicates, and duplicates do contribute to SUM. On the other hand, swapping MIN and MAX seems to be OK. However, when you try to prove equation (b), you will notice that it is actually false, too.

   (a) Let R(a,b) = { (1,1), (1,1), (2,2) }.

   $$\begin{aligned} LHS &= \gamma_{MIN(a) \rightarrow y, x}(\gamma_{a, SUM(b) \rightarrow x}(R)) \\ &= \gamma_{MIN(a) \rightarrow y, x}(\{(1,2),(2,2)\}) \\ &= \{(1,2)\} \\ RHS &= \gamma_{y, SUM(b) \rightarrow x}(\gamma_{MIN(a) \rightarrow y, b}(R)) \\ &= \gamma_{y, SUM(b) \rightarrow x}(\{(1,1),(2,2)\}) \\ &= \{(1,1),(2,2)\} \end{aligned}$$

   Since $LHS \neq RHS$, equation (a) is false.

   (b) Let R(a,b) = { (1,4), (1,3), (2,3) }.

   $$\begin{aligned} LHS &= \gamma_{MIN(a) \rightarrow y, x}(\gamma_{a, MAX(b) \rightarrow x}(R)) \\ &= \gamma_{MIN(a) \rightarrow y, x}(\{(1,4),(2,3)\}) \\ &= \{(1,4),(2,3)\} \\ RHS &= \gamma_{y, MAX(b) \rightarrow x}(\gamma_{MIN(a) \rightarrow y, b}(R)) \\ &= \gamma_{y, MAX(b) \rightarrow x}(\{(1,4),(1,3)\}) \\ &= \{(1,4)\} \end{aligned}$$

   Since $LHS \neq RHS$, equation (b) is false.

4. Exercise 16.5.1

   Simple estimation gives us

$$Est_s = \frac{T(R)T(S)}{MAX(V(R,Y),V(S,Y))} = \frac{52 \times 78}{20} = 203$$

   With the histograms, based on the estimation method in Example 16.27, we have

$$Est_h = 5 \times 10 + 6 \times 8 + 4 \times 5 + 5 \times 3 + 7 \times 2 + 15 \times 2 \times 3 = 237$$

   The two estimates are actually quite close, but note that the confidence of Est_h is higher because we know for sure that the join size is at least 118.

5. Exercise 18.2.4 (e)

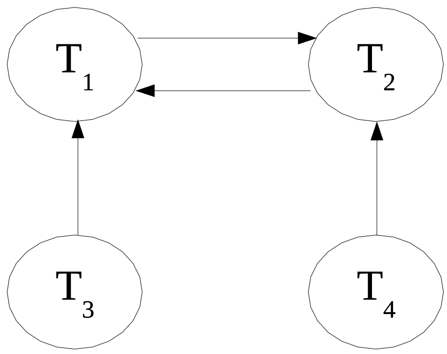

Figure 1: Precedence Graph

   This schedule is not conflict-serializable, or serializable for that matter.

6. Exercise 18.4.2 (b)

   Three. There are still two interleavings that are equivalent to (T_1,T_2) (see the online solution for the (a) part of the exercise), but there is only one interleaving that is equivalent to (T_2,T_1).

7. Exercise 18.7.3

Based on the locking order, we have the following diagram:

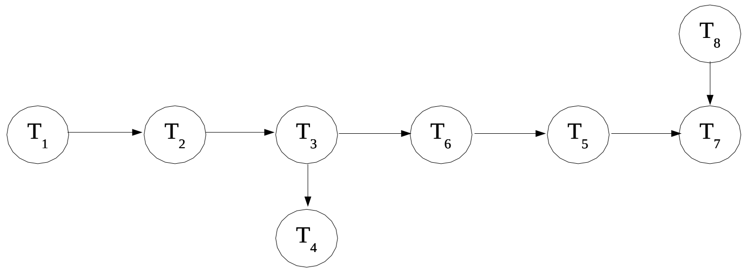

Figure 2: Locking Order

If we remove T_8, we have four serial orders:

(a) T_1, T_2, T_3, T_4, T_6, T_5, T_7

(b) T_1, T_2, T_3, T_6, T_4, T_5, T_7

(c) T_1, T_2, T_3, T_6, T_5, T_4, T_7

(d) T_1, T_2, T_3, T_6, T_5, T_7, T_4

Since T_8 must come before T_7, we have $7 + 7 + 7 + 6$ ways to add back T_8, therefore there are total of 27 serial orders.