Database Systems, CSCI 4380-01
Homework # 8
Due Thursday May 3, 2018 at 11 PM

Homework Statement.

This homework is worth 6% of your total grade. If you choose to skip it, Final Exam will be worth 6% more.

This homework is on query cost estimation, transaction concurrency, durability and tuning. Please type your answers in a PDF document and submit on Gradescope. No handwritten homeworks please.

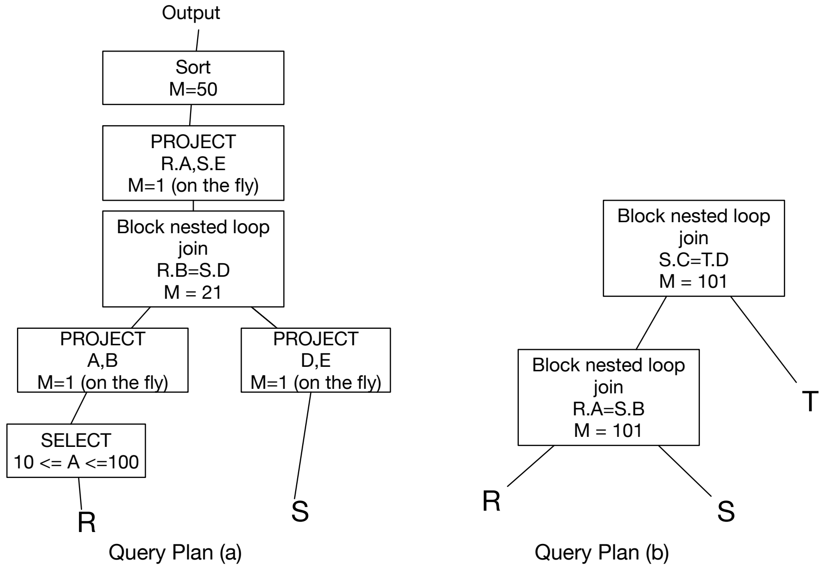

Figure 1: Query trees for Question 1

Question 1 (30 points). Estimate the cost of the query trees in Figure 1 given the statistics below. Show all your work.

A disk page and memory block is 8KB total, use 8,000 bytes for simplicity.

Relation `R(A,B,C,D)`: Pages(R)=20,000, Tuples(R)=400,000. Attribute A takes 4 bytes and B takes 20 bytes to store.

Values(R.A)=2,000 where Minval(R.A)=1 and Maxval(R.A)=2,000.

Values(R.B)=40,000

Relation `S(B,C,D,E,F,G)`: Pages(S)=80,000, Tuples(S)=1,000,000. Attribute D tables 20 bytes to store and E takes 40 bytes to store.

Values(S.D)=30,000, Values(S.C)=100,000, Values(S.B)=10,000.

Relation `T`: Pages(T)=10,000, Tuples(T)=100,000, Values(T.D)=20,000.

Question 2 (20 points). Suppose after a crash, you found the following log entries on disk. Furthermore, you are also for each data page on disk, the LSN of the last recorded log change on that page.

Discuss where the analysis step starts from and what its result is.

Then, discuss which operations will be checked for a REDO, and which update operations need to be redone (and in which order).

Finally, show which operations need to be undone (and in which order).

Show enough detail to explain how you arrived at your solutions.

```
LOG:
LSN   LOG ENTRY             PREVLSN
101   T1 update P1 10 20     -
102   T3 update P3 A B       -
103   commit T3              102
104   T2 update P5 XY ZZ     -
105   T4 update P3 B C       -
106   T2 update P4 5 8       104
107   rollback T2            106
108   checkpoint
109   TT:  T2 107, T4 105, T1 101
110   DPT: P4 106, P3 105
111   end checkpoint
112   T2 undo 106 (P4 8 5)   104
113   T5 update P4 5 12      -
114   T5 update P6 X1 Y2     113
115   T1 update P1 20 30     101
116   commit T5              114
117   T1 update P5 ZZ RT     115
118   T7 update P4 12 15     -
119   T6 update P2 dd aa     -
120   commit T1              117
121   T7 update P1  30 40    118
122   T8 update P6 Y2 Z3     -

DATA PAGES:

PAGE ID  LSN OF LAST UPDATE
P1       115
P2       119
P3       105
P4       113
P5       117
P6       122
```

Question 3 (30 points). You are given the following three different (possible) schedules for the same four transactions. For each schedule, answer the following.

$$\begin{aligned}
S1 &= r_1(Z)\ w_1(Z)\ r_1(Y)\ r_2(X)\ r_3(Z)\ r_3(X)\ w_1(Y)\ r_2(Y)\ w_3(X)\ r_4(K) \\
&\quad w_3(Z)\ r_4(X)\ w_4(K)\ w_2(Y) \\
S2 &= r_1(Z)\ w_1(Z)\ r_1(Y)\ r_4(K)\ r_4(X)\ r_2(X)\ r_3(Z)\ r_3(X)\ w_4(K)\ w_1(Y) \\
&\quad r_2(Y)\ w_3(X)\ w_3(Z)\ w_2(Y) \\
S3 &= r_1(Z)\ w_1(Z)\ r_1(Y)\ r_4(K)\ r_4(X)\ w_3(X)\ w_3(Z)\ r_2(X)\ r_3(Z) \\
&\quad r_3(X)\ w_4(K)\ w_1(Y)\ r_2(Y)\ w_2(Y)
\end{aligned}$$

1. List all conflicts and draw the conflict graph.
2. Based on your conflict graph, discuss whether it is serializable or not, and why.
3. Discuss whether it is possible to obtain this schedule if two phase locking is used? Discuss why or why not.

Question 4 (20 points). For this last question of the homework, your job is to optimize at least one of the queries given in the answer to Homework #5.

To optimize a query, first you will look at its estimated cost. To do this, you can simply add the word `explain` to the beginning of the query. For example:

```
olympics=> explain select * from summer_medals where id = 5 ;
                                       QUERY PLAN
-----------------------------------------------------------------------------------------
 Index Scan using summer_medals_pkey on summer_medals  (cost=0.29..8.30 rows=1 width=22)
   Index Cond: (id = 5)
(2 rows)

olympics=> explain select * from summer_medals where oid = 5 ;
                           QUERY PLAN
----------------------------------------------------------------
 Seq Scan on summer_medals  (cost=0.00..588.56 rows=1 width=22)
   Filter: (oid = 5)
(2 rows)


olympics=> explain select * from summer_medals where oid =5 order by medal;
                              QUERY PLAN
----------------------------------------------------------------------
 Sort  (cost=588.57..588.58 rows=1 width=22)
   Sort Key: medal
   ->  Seq Scan on summer_medals  (cost=0.00..588.56 rows=1 width=22)
         Filter: (oid = 5)
(4 rows)
```

In each query plan, you get two estimated costs (e.g. `0.29 and 0.83`). The first one is the time to the first answer (which is non-zero given the unavoidable cost of scanning the index first) and the second one is the cost of getting all the answers. In the second query, the initial cost is zero because you can start to produce tuples as soon as you start scanning the relation (assuming you find some matching tuples). You can see that the cost of the third query is the same for both first and all results because sort is a blocking query, you cannot return any results until the sort is complete. Then, you can return all the results.

In this question, you are asked to improve the full running time (the second number) of one of the queries by either rewriting the query or by introducing an index. To show the result, you must document the full query plan before your change and the one after in your answer.

To allow you to run this part, you must create the hw#5 data on your computer. I am happy to do this for you on the DB server as well. Expect this to be available by the end of the weekend. In the meantime, you can work on improvements by query rewriting or think of good indices. We will discuss this topic on monday in more detail.

Who will have the best improvement? Technically we only need one. But if you want to show us more than one, feel free.