Database Systems, CSCI 4380-01
Homework # 8
Due Thursday December 9, 2021 at 11:59:59 PM

Homework Statement. This homework is worth 5% of your total grade. If you choose to skip it, Final Exam will be worth 5% more. You are required to complete at least half the questions in this homework.

Question 1. You are given the following information:

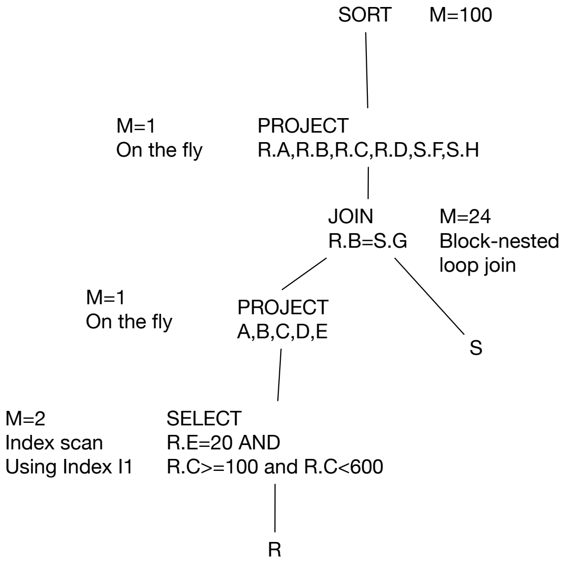
```
PAGES(R) = 1000, TUPLES(R) = 80,000
PAGES(S) = 4000, TUPLES(S) = 600,000

Index I1 on R(E,C,A,B,D) with 500 pages at the leaf
    level with 3 levels (root, internal, leaf)

VALUES(R.A) = 80,000 [0 (minval) to 80,000 (maxval)]
VALUES(R.B) = 10,000 [0 (minval) to 200,000 (maxval)]
VALUES(R.C) = 5,000 [0 (minval) to 10,000 (maxval)]
VALUES(R.D) = 200 [0 (minval) to 200 (maxval)]
VALUES(R.E) = 5 [0 (minval) to 500 (maxval)]

VALUES(S.F) = 70,000 [0 (minval) to 80,000 (maxval)]
VALUES(S.G) = 5,000 [0 (minval) to 200,000 (maxval)]
VALUES(S.H) = 800 [0 (minval) to 1000 (maxval)]

Assume: each attribute above is 8 bytes long, the
    usable space on a data page (or memory block) is
    6000 bytes (after taking header information out).
```



Given the following query:

```
SELECT R.A, R.B, R.C, R.D, R.E FROM R,S
WHERE R.B=S.G AND R.E=20 AND R.C>=100 and R.C<600
```

Based on the query plan given in the figure for this query, answer the following questions. Assume each operation pipelines the results to the next operation as in Lecture 22 Exercise.

For each part, write some brief explanation of how you computed your answer. In all your computations, use the size estimation methods we learnt in class for understanding how many tuples to expected in the output. Given the size of the resulting tuples (based on how many attributes they have), you can find out how many blocks are needed to store these tuples in memory.

(a) What is the cost of index search?

(b) What is the additional cost of the block-nested join on top of the index search?

(c) What is the additional cost of the sort on top of the cost of index search and join?

(d) What is the total cost of this query plan?

Question 2. You are given the following statistics:

```
PAGES(R) = 200
PAGES(S) = 1,200
PAGES(T) = 4,000
PAGES(R join S) = 8,000
PAGES(R join T) = 2,000
```

What is the cost of the following query plans, assuming that (a) each join uses block-nested loop join, (b) the output of the lower join is pipelined into the upper join, and (3) each join has `M=201` blocks of memory allocated?

1. `(R join S) join T`
2. `(R join T) join S`

Question 3. You are given the following schedule: (a) list all conflicts, (b) check if the schedule is serializable by drawing the conflict graph, and (c) discuss if it is possible to obtain this schedule using Two Phase Locking.

`S: r1(x) r2(z) r1(y) w2(w) w2(z) r3(z) w3(x) r1(w) w1(y) w3(z)`

Question 4. In this question, you are asked the reverse of previous homework. Improve the expected run time of **two queries** that are given in this homework (based on my own answers to queries in homeworks 4 and 5).

To facilitate this, I have created a new database for each student with name `db2_<username>`. This is a larger database than the one you used in Homework #6. Please do not drop any tables as in Homework #6.

You can find the expected run time of a query using `EXPLAIN` before the query:

```
radiodb => explain select songid from spotify;
                         QUERY PLAN
--------------------------------------------------------------------
 Seq Scan on spotify (cost=0.00..445.93 rows=27193 width=8)
(1 row)


radiodb => explain select songid, count(*) as num
radiodb -> from spotify group by songid order by num desc;
                               QUERY PLAN
---------------------------------------------------------------------------------
 Sort  (cost=594.35..594.98 rows=250 width=16)
   Sort Key: (count(*)) DESC
   ->  HashAggregate (cost=581.89..584.39 rows=250 width=16)
         Group Key: songid
         ->  Seq Scan on spotify (cost=0.00..445.93 rows=27193 width=8)
(5 rows)
```

The first cost value is the estimated cost to get the first tuple and the second cost value is the estimated cost to get all the tuples. We only care about the second cost. In queries involving group by or sort, both values are roughly the same because you cannot get any tuples until the sorting is mostly complete.

You can improve the run time in one of two ways:

- Create a single index and check if the cost has improved. To document the improvement, list (a) the query text, (b) the first 3 lines of the query plan before you create the index, (c) the index creation command, and (d) full query plan after you create the index. It is important that the full query plan in (d) shows that your index is being used.
- Alternatively, you can rewrite the query solution such that it is different than your previous solution to the same query and different than the solution I provided in this homework.

  Document that the rewritten query has lower query cost by listing (a) old query text, (b) the first 3 lines of the old query plan, (c) the new query text, and (d) the new query plan.

In either case, the query plans after your changes should be lower than before for the second cost value (the time to get all the answers). However, it is not required that the reduction in cost is large. Even smaller improvements will be accepted.

When creating indices, consider a few simple rules of thumb that you may have learn from the previous homework:

- Relations with lots of tuples and span many disk pages are likely to benefit more from indices.
- Conditions that are more selective are likely to be more useful. In this include any join conditions as well.
- In case very selective conditions don't exist, you can still target index only scans. In this case, ordering of attributes is important.

When I tested, I found small improvements with indices in many queries.

Please document this for two queries! At least one query should use an index for its improvement.

SUBMISSION INSTRUCTIONS.

Submit a single text or PDF file that documents your answers to the homework.