# CS3223 Tutorial 6: Query Optimization & Serializability

**Wk 9, Sem 2, 2022/23**

**Questions to be discussed: 1 & 4**

1. Consider a database with the following two relations where the key attributes are shown underlined:

   - R (<u>a</u>, b, c, d)
   - T (<u>h</u>, i, j)

   Assume the following for this question.

   1. Relation `R` contains 10,000 pages with each page containing 30 records.
   2. Relation `T` contains 8,000 pages with each page containing 100 records.
   3. There are three unclustered indexes on relation `R`:
      - $I_b$: a B$^+$-tree index on (b) with at most 100 entries in each leaf page.
      - $I_c$: a B$^+$-tree index on (c) with at most 100 entries in each leaf page.
      - $I_{bc}$: a B$^+$-tree index on (b, c) with at most 50 entries in each leaf page.
   4. There are three unclustered indexes on relation `T`:
      - $I_i$: a B$^+$-tree index on (i) with at most 200 entries in each leaf page.
      - $I_j$: a B$^+$-tree index on (j) with at most 200 entries in each leaf page.
      - $I_{ij}$: a B$^+$-tree index on (i, j) with at most 100 entries in each leaf page.
   5. Each of the indexes has two levels of internal nodes.
   6. Only 10% of `R` records satisfy the condition "b > 20".
   7. Only 5% of `R` records satisfy the condition "c = 100".
   8. Only 1% of `R` records satisfy both the conditions "b > 20" and "c = 100".
   9. Only 5% of `T` records satisfy the condition "i > 50".
   10. Only 5% of `T` records satisfy the condition "j > 30".
   11. Only 4% of `T` records satisfy both the conditions "i > 50" and "j > 30".
   12. The cost metric to use is the number of page I/Os. Ignore the cost of writing out the final result.
   13. There are 25 buffer pages available.
   14. The database system supports only four join algorithms (Block Nested-loop Join, Indexed Nested-loop Join, Optimized Sort-Merge Join, and Grace Hash Join), and supports only the hash-based algorithm for set intersections.

   Consider the following three queries.

   ```
   Q1: SELECT  *
       FROM    R
       WHERE   b > 20
       AND     c = 100
   ```

   ```
   Q2: SELECT  *
       FROM    T
       WHERE   i > 50
       AND     j > 30
   ```

   ```
   Q3: SELECT  *
       FROM    R, T
       WHERE   R.d = T.h
       AND     R.b > 20
       AND     R.c = 100
       AND     T.i > 50
       AND     T.j > 30
   ```

   Answer the following three questions:

   (a) What is the least cost plan for query $Q1$? What is its cost?

   (b) What is the least cost plan for query $Q2$? What is its cost?

   (c) What is the least cost plan for query $Q3$? What is its cost?

2. Consider a relation $R(\underline{a}, b, c)$, where the domains of all the attributes are positive integers. Assume that $||R|| = 10000$, $||\pi_b(R)|| = 100$, and $||\pi_c(R)|| = 20$.

   Estimate the result size for each of the following queries.

   (a) SELECT * FROM R WHERE $b = 10$

   (b) SELECT * FROM R WHERE $(b \geq 20)$ AND $(b < 40)$

   (c) SELECT * FROM R WHERE $b \neq 7$

   (d) SELECT * FROM R WHERE $(b = 20)$ AND $(c = 40)$

   (e) SELECT * FROM R WHERE $(b = 20)$ OR $(c = 40)$

3. Consider a database with the following three relations:

   - $R(a, b, c)$ with $||R|| = 200$, $||\pi_b(R)|| = 20$, and $||\pi_c(R)|| = 50$
   - $S(d, e, b)$ with $||S|| = 800$ and $||\pi_b(S)|| = 40$
   - $T(f, g, c)$ with $||T|| = 500$ and $||\pi_c(T)|| = 100$

   Estimate the result cardinality for each of the following queries:

   (a) $Q_1$: SELECT * FROM R JOIN S ON R.b = S.b

   (b) $Q_2$: SELECT * FROM R JOIN S ON R.b = S.b JOIN T ON R.c = T.c

4. Consider a relation $R$ with $||R|| = 121$ and an attribute $A$ with $||\pi_A(R)|| = 20$. The actual distribution of attribute $A$ is shown below.

| **Value of A** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **# of tuples** | 1 | 15 | 6 | 8 | 2 | 3 | 0 | 1 | 20 | 3 | 0 | 10 | 6 | 8 | 12 | 7 | 0 | 8 | 5 | 6 |

   (a) Construct an equidepth histogram $H_3$ with 3 buckets.

   (b) Estimate the size for each of the following queries using $H_3$.

      i. $Q_1$: SELECT * FROM R WHERE A = 5

      ii. $Q_2$: SELECT * FROM R WHERE A = 8

      iii. $Q_3$: SELECT * FROM R WHERE A $\geq$ 6 AND A $\leq$ 17

   (c) Construct an equidepth histogram $H_3'$ with 3 buckets and top-2 MCV.

   (d) Repeat part (b) using $H_3'$.

   (e) Construct an equidepth histogram $H_5$ with 5 buckets.

   (f) Repeat part (b) using $H_5$.

5. (Exercise 17.2, R&G) For each of the following schedules, state whether is it view/conflict serializable.
   (a) $R_1$(X), $R_2$(X), $W_1$(X), $W_2$(X), $Commit_1$, $Commit_2$
   (b) $W_1$(X), $R_2$(Y), $R_1$(Y), $R_2$(X), $Commit_1$, $Commit_2$
   (c) $R_1$(X), $R_2$(Y), $W_3$(X), $R_2$(X), $R_1$(Y), $Commit_1$, $Commit_2$, $Commit_3$
   (d) $R_1$(X), $R_1$(Y), $W_1$(X), $R_2$(Y), $W_3$(Y), $W_1$(X), $R_2$(Y), $Commit_1$, $Commit_2$, $Commit_3$
   (e) $R_1$(X), $W_2$(X), $W_1$(X), $Commit_2$, $Commit_1$
   (f) $W_1$(X), $R_2$(X), $W_1$(X), $Commit_2$, $Commit_1$
   (g) $R_2$(X), $W_3$(X), $Commit_3$, $W_1$(Y), $Commit_1$, $R_2$(Y), $W_2$(Z), $Commit_2$
   (h) $R_1$(X), $W_2$(X), $Commit_2$, $W_1$(X), $Commit_1$, $R_3$(X), $Commit_3$
   (i) $R_1$(X), $W_2$(X), $W_1$(X), $R_3$(X), $Commit_1$, $Commit_2$, $Commit_3$
   (j) $R_1$(X), $R_2$(Y), $W_3$(X), $W_3$(Z), $R_2$(X), $R_1$(Y), $W_1$(Z), $W_2$(Z), $Commit_1$, $Commit_2$, $Commit_3$
6. Prove that a conflict serializable schedule is also a view serializable schedule.
7. Prove that a schedule is conflict serializable if and only if its conflict serializability graph is acyclic.
8. Prove that a view serializable schedule without any blind write is also a conflict serializable schedule.