MSBD 6000L: Database Systems

Assignment 3: Indexing, Query Processing and Transactions

Assigned: November 1, 2022 Value: 10% of course grade
Due: 23:00 (11:00 p.m.), November 26, 2022

IMPORTANT REMINDER

This is an individual assignment. What you submit should be *your own work*. While you may discuss general assignment issues with other students, you are not allowed to collaborate with other students (past or present), to develop common answers, to share answers or to copy someone else's answer. Copying, sharing or collaborating will be severely penalized. All those involved in a copying/sharing/collaborating incident will automatically receive a grade of 0 and may be reported for further disciplinary action.

1. A company wants to develop a database for its 30,000 employees. A page is 512 bytes, a pointer to a page is 6 bytes and a pointer to a record is 7 bytes. There are six fields in the Employee file: empNo (4 bytes), name (30 bytes), address (40 bytes), phoneNo (8 bytes), email (30 bytes) and hkid (8 bytes). The Employee file is ordered according to the primary key empNo.

   60% of queries to the Employee file involve retrieval according to empNo, 30% according to hkid and 10% according to the remaining fields.

   Given that you are restricted to constructing only one single-level index, would you construct a primary index on empNo or a secondary index on hkid? Your goal is to minimize the number of page I/Os for the given types of queries.

   Justify your answer by showing the overall average page I/O cost of querying the Employee file using

   a) no index.
   b) a primary single-level index on empNo.
   c) a secondary single-level index on hkid.

   Show clearly your calculations. Answers with no or unclear calculations will incur a 50% penalty.

2. For the Employee file of Question 1, suppose that you can construct a B^+-tree index instead of a single-level index. Assume that all nodes (both leaf and internal) of the B^+-tree have *minimum occupancy*. What would be the overall average page I/O cost of querying the Employee file using

   a) a clustering B^+-tree index on the primary key empNo.
   b) a non-clustering B^+-tree index on hkid.

   In addition to calculating the overall average page I/O cost of querying the Employee file using each index, also show for each index your calculations for

   i. the number of values and pointers in the leaf and internal nodes.
   ii. the height of the B^+-tree.

   Show clearly your calculations. Answers with no or unclear calculations will incur a 50% penalty.

3. For the query $\pi_{A,B,C,D}(R \bowtie_{A=C} S)$, assume the following:
   - R is 10 pages; each R tuple is 300 bytes.
   - S is 100 pages; each S tuple is 500 bytes.
   - The combined size of attributes A, B, C and D is 450 bytes.
   - A and B are in R and have combined size of 200 bytes; C and D are in S.
   - A is a key for R.
   - Each S tuple joins with exactly one R tuple.
   - The page size is 1024 bytes.
   - The buffer size *M* is 3 pages.

a) Show what are the *minimum* page I/O costs if the join uses the (optimized) block nested-loop join method.
   i. Page I/O cost for join eliminating all unwanted attributes during the join.
   ii. Page I/O cost for projection using external sorting and removing duplicate tuples on-the-fly during the merge passes.
   iii. Total query processing page I/O cost.

b) Show what are the *minimum* page I/O costs if the join uses the merge join method.
   i. Page I/O cost to sort R.
   ii. Page I/O cost to sort S.
   iii. Page I/O cost to join R and S.
   iv. Total query processing page I/O cost.

4. Consider the following two transactions:

T_1
read(A)
read(B)
if A = 0 then B := B + 1
write(B)

T_2
read(B)
read(A)
if B = 0 then A := A + 1
write(A)

Let the consistency requirement after both transactions execute successfully be A=0 ∨ B=0, with A=B=0 the initial values.

a) Show that every serial execution involving these two transactions preserves the consistency of the database.

b) Give a concurrent execution of T_1 and T_2 that produces a non-serializable schedule and show why the schedule is non-serializable.

c) Is there a concurrent execution of T_1 and T_2 that produces a serializable schedule? Explain your answer.

WHAT TO SUBMIT

Submit your answers to the questions as a pdf file named Assignment3.pdf.

HOW TO SUBMIT

By 11:00 p.m. on Saturday, November 26, upload your Assignment3.pdf file to Canvas by selecting *Assignment 3* in the Assignments section under the Assignments tab of Canvas, and then selecting the Submit Assignment button. To check your submission, select the Submission Details button. For help, select the Help button.

You are responsible to ensure that your submission is correctly uploaded to Canvas.

Under no circumstances will late submissions or submissions by email be accepted.

GRADING

Item	Value
Question 1	20%
Question 2	20%
Question 3	40%
Question 4	20%