National University of Computer and Emerging Sciences
Lahore Campus

Advance Database Concepts (CS4064)

Assignment 3 Total Marks: 150

_____ _____ _____
Name Roll No. Section

Do not write below this line.

Note: Please ensure that you attempt all questions and their respective parts in the given order.

Q. No 1: Consider the following part of library database schema and the query in *SQL* and *RA*:

Book (BookID, Title, Category, Publisher, PublishYear)
Author (AuthorID, AuthorName, Gender, Email, OriginCity)
BookAuthor (BookID, AuthorID)

```
SELECT Title, AuthorName, Publisher
FROM Author A JOIN BookAuthor BA ON A.AuthorID=BA.AuthorID JOIN Book B ON B.BookID=BA.BookID
WHERE Gender= 'Male' AND Category= 'Education';
```

$\pi_{Title, AuthorName, Publisher}(\sigma_{Gender='Male' \wedge Category='Education'}(Author * BookAuthor * Book))$

Your task is to optimize this query and draw the best possible query tree for this query. Take appropriate database statistics to support your answer. (Show all steps) [10]

CLO # 3: To develop a solution for given scenario/challenging problem in the domain of DB systems.

Q. No 2: [5+5]

a. Consider the above library database schema and the query in *SQL/RA* given in *Q#1*. Assume that the frequency of access of this query is very high. Which attributes are more appropriate to create indexes for an efficient execution strategy to improve the performance of this query?

b. Consider the above library database schema and assume that *Book, Author* and *BookAuthor* tables have *100000*, *100000* and *50000* rows respectively. Estimate the potential *Join Cardinality* (*jc*) of *Book* $\bowtie_{BookID=BookID}$ *BookAuthor* (i.e., max number of rows resulting from the inner join of these two tables). Justify your answer.

CLO # 3: To develop a solution for given scenario/challenging problem in the domain of DB systems.

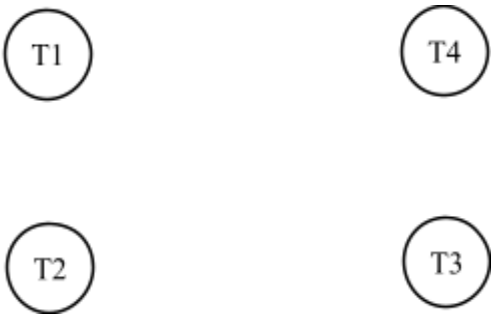
Q. No 3: Assume: A block size is *B= 1024* bytes, file has *r= 1,000,000* records, each record is *100* bytes long, a block pointer is *P= 10* bytes, a record pointer is P_R*= 11* bytes, and a key field for the index is 6 bytes long. A database system uses a B+-trees index on a key field. A leaf node and non-leaf node are one block in size and contain as many keys (and appropriate pointers) as will fit in a block. How many blocks will this index use? Also estimate the number of block accesses needed to search for and retrieve a record from the file given its key value using the B+-tree index. Show your working. [10]

CLO # 2: Apply the models and approaches to become enabled to select and apply appropriate methods for a particular case.

Q. No 4: Consider the following schedule: [5]

S: r1(X), r2(Z), w1(Z), r3(X), r3(Y), w1(X), w3(Y), r4(Y), w4(Z), w4(Y).

Draw the serializability (precedence) graph for this schedule. State whether this schedule is conflict-serializable (correct) or not. If the schedule is conflict-serializable, write down the equivalent serial schedule(s) otherwise explain why it is not. *Also state whether this schedule is view-serializable or not.*



CLO # 2: Apply the models and approaches to become enabled to select and apply appropriate methods for a particular case.

Q. No 5: Consider the following schedule of actions: [10+10]

S: r1(X), r2(Z), w1(Z), r3(X), r3(Y), w1(X), w3(Y), r2(Y), c3, c2, c1, w4(Z), w4(Y), c4.

For each of the following concurrency control mechanisms, describe how the concurrency control mechanism handles the schedule. Assume that the timestamp of transaction *Ti* is *i*. For lock-based concurrency control mechanisms, add lock and unlock requests to the above schedule of actions as per the locking protocol. The DBMS processes actions in the order shown. If a transaction is blocked, assume that all its actions are queued until it is resumed; the DBMS continues with the next action (according to the listed schedule) of an unblocked transaction.

a. Rigorous 2PL with timestamps used for deadlock detection (Use wait-for-graph to deal with deadlock)
b. Basic Timestamp Ordering (Assume T1 < T2 < T3)

a)

T_1	T_2	T_3	T_4

Wait-for-graph:

b)

T1	T2	T3	T4	X		Y		Z	
				RTS	*WTS*	*RTS*	*WTS*	*RTS*	*WTS*

CLO # 2: Apply the models and approaches to become enabled to select and apply appropriate methods for a particular case.

Q. No 6: Consider the following log at the point of system crash. Suppose that we use ARIES recovery algorithm to answer the following questions. [9]

LSN	Last_LSN	Trans_ID	Type	Page_ID	Other_Info
1	0	T1	Update	A	…
2	0	T2	Update	C	…
3	1	T1	Commit		…
4	2	T2	Update	A	…
5	begin checkpoint				
6	end checkpoint				
7	4	T2	Commit		…
8	0	T3	Update	B	…
9	0	T4	Update	C	…
10	8	T3	Update	A	…
11	9	T4	Commit		…

a. Show the contents of transaction and dirty page table at the time of checkpoint. [2]

Trans_ID	LSN	Status

Page_ID	LSN

b. What is done during Analysis? Be precise about the points at which Analysis begins and ends and show the contents of transaction and dirty page table reconstructed in this phase. [3]

Analysis Phase: **From:** **To:**

Transaction Table

Trans_ID	LSN	Status

Page_ID	LSN

Dirty Page Table

c. What is done during Redo? Be precise about the points at which Redo begins and ends. [2]

d. What is done during Undo? Be precise about the points at which Undo begins and ends. [2]

Q. No 7: Consider the bank database, and the following SQL query: [5+5+5]

Customer (custID, custName, cnic, birthDate, address, ...)
Account(accNo, custID, accTitle, accType, openingDate, ...)
Transaction(tID, accNo, transType, amount**,** transDate, ...)

Write an efficient relational-algebra expression that is equivalent to these queries and draw the optimal query plan for this query.

1. **SELECT** C.cnic, A.accNo, A.Title, T.noOfTrans **FROM** customer C **JOIN** account A **ON** C.custID=A.custID **JOIN** (SELECT accNo, COUNT(*) **AS** noOfTrans FROM transaction **GROUP BY** accNo) T **ON** A.accNo=T.accNo

2. **SELECT** accNo **FROM** Account **WHERE** accType=’Saving’ **AND** openingDate=’12-02-2024’

3. **SELECT** accNo **FROM** Transaction **WHERE** amount >25000 **OR** transDate =’12-02-2024’

Q. No. 8: Assume that the number of buffers available in main memory for implementing the join is k= n_b = 5 blocks. The DEPARTMENT file consist of r_D = 40 record stored in b_D = 10 disk blocks and that the EMPLOYEE file consists of r_E = 600 record stored in b_E = 200 disk blocks. [3*5=15]

Suppose that secondary indexes exist on both the attributes
Ssn of employee and *Mgr_ssn* of department,
with the number of index levels X_{Ssn} = 4 and X_{Mgr_ssn} = 2, respectively.

DEPARTMENT $\bowtie_{Mgr\_ssn=Ssn}$ **EMPLOYEE**

Apply below Joining Algorithms and find cost estimation. Suggest what will be the best joining algorithm to apply on above case.

1. **Nested-loop join (nested-block join)**

2. **Index-based nested-loop join**

3. **Sort-merge join**

4. **Partition-hash join**

Conclusion:

Q. No. 9: Write down the Cost function of the following selection operations [1*5=5]

1. Primary index to retrieve a single record

2. A hash key to retrieve a single record

3. An ordering index to retrieve multiple records

4. A clustering index to retrieve multiple records

5. Secondary(B+-tree) index in worst case as well as for range queries

Now consider following Statistics of EMPLOYEE [4*8=32]

Suppose that the EMPLOYEE file has r_E = 1000 records stored in b_E = 2,00 disk blocks with blocking factor bfr_E = 5 records/block and the following access paths:

1. A clustering index on Salary, with levels x_{Salary} = 3 and average selection cardinality s_{Salary} = 20.

   Find a selectivity of sl_{Salary} ?

2. A secondary index on the key attribute Ssn, with xSsn = 4 and average selection cardinality s_{Ssn} = 1, what would be the $sl_{Ssn=?}$

3. A secondary index on the nonkey attribute Dno, with x_{Dno} = 2 and first-level index blocks b_{I1Dno} = 4. There are *NDV (Dno, EMPLOYEE)* = 125 distinct values for Dno, so what would be the the selectivity of Dno is sl_{Dno} = ? and the selection cardinalty is s_{Dno} = ?

4. A secondary index on Sex, with x_{Sex} = 1. There are *NDV (Sex, EMPLOYEE)* = 2 values for the Sex attribute, so the average selection cardinality is s_{Sex} =?

The use of cost functions with the following examples and suggest the best cost function for each operation.

OP1: σSsn='123456789' (EMPLOYEE)

OP2: σDno > 5 (EMPLOYEE)

OP3: σDno = 5 (EMPLOYEE)

OP4: σDno=5 AND SALARY>30000 AND Sex='F' (EMPLOYEE)

Q. No. 10: Assume that the number of buffers available in main memory for implementing the join is **no = 3 blocks**. The DEPARTMENT file consists of **r(D) = 50 records** stored in **b(D) = 10** disk blocks and the Student file consists of **r(S)= 6000 records** stored in **b(S) = 2000 disk blocks.** **[1+(3*6)=19]**

a) How many total number blocked fetched by using Nested loop join strategy if we performed following query [1]

Select * from Department D join Student S ON D.DID=S.DID

Total Blocks=

b) **What would be Sort-merge join cost if:**

1. If both student and Department are already sorted

2. If both are not sorted and sorted cost using external sort-merge algorithm is (b loga b) b is the total number of blocks in a file

c) Suppose the Student table has 80 distinct values over the attribute DID and the Department has 50 distinct values over the attributa DID.

Select * from Department D join Student S ON D.DID=S.DID

1. Find join selectivity ratio of both tables

   Js=

2. Find join cardinality of both tables

   Jc=

Select * from Department D where D.DID NOT IN (Select S.DID from Student S)

3. Find join selectivity ratio of both tables

   Js=

4. Find join cardinality of both tables

   Jc=