Name: ID: Section:

**General Remarks:**

1. This is a closed book exam.
2. Attempt all questions.
3. Show all your work. Partial credit is possible for correct answers.
4. State any necessary assumptions if you did.

|  |  |  |
| --- | --- | --- |
| Question | Max Mark | Mark |
| 1 | 5 |  |
| 2 | 5 |  |
| 3 | 8 |  |
| 4 | 9 |  |
| 5 | 4 |  |
| 6 | 6 |  |
| 7 | 4 |  |
| 8 | 4 |  |
| Total | **45** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Outcome | | Question | Max | Mark |
| B | Apply the concepts of file structures and calculate access time and disk capacity. | 1 | 5 |  |
| C | Analyze the indexing methods and the use of B-trees and hash tables in the design of database systems. | 2, 3 | 13 |  |
| D | Evaluate and discuss the query processing and the choices made by the query optimizer, and database tuning | 4 | 9 |  |
| E | Analyze transaction sequence processing, and recovery and concurrency control methods | 5, 6, 7, 8 | 18 |  |

**Question 1:** [5 marks] File structures   
[Outcome B]

* + Assume records have fixed length,
    1. Size of a record = 150 bytes
  + Disk characteristics:
    1. Number of bytes per sector = 512
    2. Number of sectors per track = 80
    3. Number of tracks per cylinder = 8
    4. Number of cylinders = 1024

[1 mark] What is the capacity of a track (size in bytes)?

**Track capacity = number of sector per track \* bytes per sector**

= 80 sectors/track \* 512 bytes/sector

**=40,950 bytes/track**

[1 mark] How many sectors are in one cylinder?

**Sectors/cylinder = No. of tracks/cylinder \* No.of sectors/ track**

= 8 \* 80

**= 640 sectors per cylinder**

[1mark] How many records can be stored per cylinder?

**Records/cylinder = Cylinder Capacity / Record Size**

= (Sectors/cylinder \* Bytes/sector) / Record Size

= (640 \* 512) / 150

**= 2048 Records per Cylinder**

[1 mark] How many cylinders are needed to store 40,000 records?

**No. of cylinders required = Total No. of records / Records/cylinder**

= 40,000 / 2048

**= 20 Cylinders**

[1 mark] What is the size in bytes of the internal fragmentation of one cylinder?

Internal fragmentation is the space that is wasted in a cylinder due to partial records.

**Internal fragmentation = Cylinder Capacity - (Records/cylinder \* Record Size)**

= (Sectors/Cylinder \* Bytes/sector) - (Records/Cylinder \* Record Size)

= (640 \* 512) - (2048 \* 150)

= 327,680 - 307,200

**= 20,480 Bytes**

**Question 2:** [5 marks] Indexing

[Outcome C]

For a multi-level index on a secondary key, non-ordering field, answer the questions below given the following specification:

* + data record size R=180 bytes
  + block size B = 1024 bytes
  + number of data records r = 25,000 records
  + index key size k = 12 bytes,
  + record pointer size PR = 8 bytes.

1. [0.5 mark] What is the blocking factor of the *data file*?

**Blocking factor = Size of block/ Record Size**

Blocking factor = 1024 / 180

**Blocking factor = 5 bytes**

1. [0.5 mark] What is the blocking factor of the *index file*?

Index file = 8+12=20

**Blocking factor = Size of block/ Entry Size**

Blocking factor = 1024 / 20

**Blocking factor = 51 bytes**

1. [1 mark] How many blocks are required to store the *data file*?

**No. of blocks = Total no of records / BF**

**No. of blocks = 25,000 / 5 = 5,000 blocks**

1. [1 mark] How many blocks are required to build the index?

Every block stores 1024 bytes / 20 bytes per index entry = 51 entries.

Total no. of Index Entries = No. of records = 25,000 Entries

So,

**Total no. of blocks = Total no. of Index Entries / Entries/block**

= 25,000 / 51

**≈ 490.20 = 491 blocks**

1. [2 mark] Create the *multi-level index* indicating the number of blocks of each level?

For creating a multi-level index, let us consider a B-tree.

Start with the root level and then create other levels until all data can be accessed. The number of levels needed depends on the no. of blocks and the branching factor of the tree.

**Suppose each node of the B-tree holds 50 entries.**

Level 0 (root level): 1 block (primarily)

Level 1: Max no. of blocks = 5000 / 50 = 100

Level 2: Max no. of blocks = 100 / 50 = 2

Level 3: Blocks are less than 50 so this is the leaf level.

**Levels of multi-level index:**

Level 0 of multi-level index (root): 1 block

Level 1 of multi-level index: 100 blocks

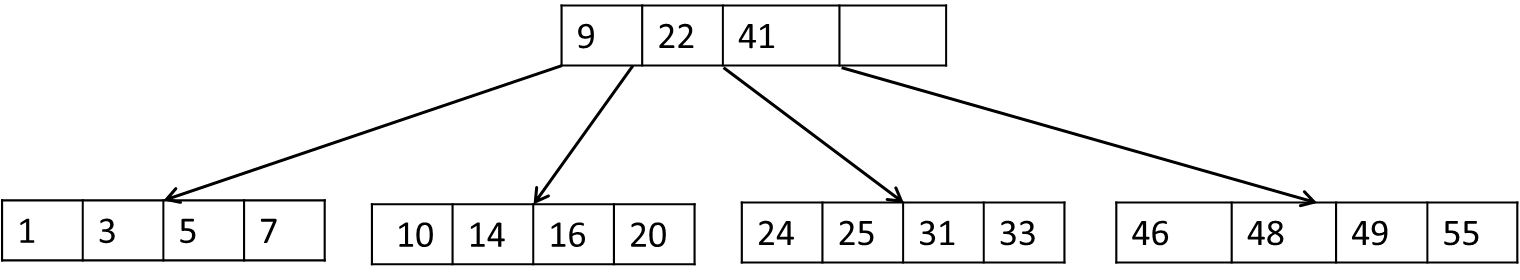
Level 2 of multi-level index: 2 blocks

**Question 3:** [8 marks] B-trees

[Outcome C]

For the given B-tree, assume that each node stores a maximum of 4 keys and a minimum of 2 keys.

1. [4 marks] Given the below B-tree, answer the following questions:



1. [1 mark] Which of the following sequence of operations will result in a tree growth?
   1. delete 20; insert 27; insert 12
   2. insert 6; insert 2; insert8
   3. delete 31; insert 6; insert 50
   4. insert 50; insert 60; insert 70
   5. none of the above

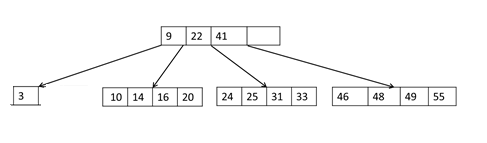
**delete 20; insert 27; insert 12**

This operation results in a tree growth because it insert a new node (27) and delete a node (20), which maintains the structure of the tree and does not results any imbalance or loss of node

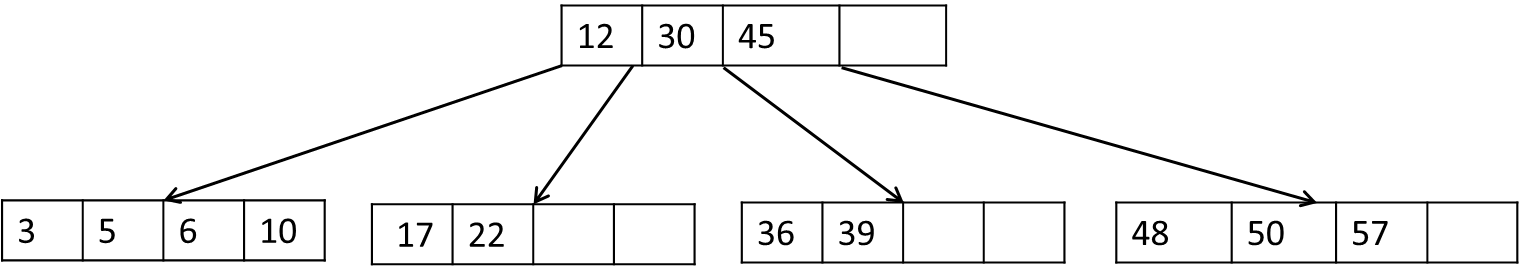
1. [1 mark] Suggest one operation that will result in changing the content of only two of the existing nodes.

Removing a key from a node that only has two keys is one operation that could change the content of the two nodes. The node removed from the tree and its neighbouring sibling may merge as a result of this deletion, altering the data of both nodes.

1. [2 marks] Redraw the B-tree after executing the following sequence of operations:   
   delete 1; delete 7; delete 5



1. [4 marks] Given the below B-tree, answer the following questions:



1. [1 mark] Which of the following sequence of operations will result in nodes merge?
   1. delete 50; delete 3; delete 10
   2. delete 50; delete 17; delete 22
   3. delete 10; delete 6; delete 22
   4. delete 10; delete 17; delete 50
   5. none of the above

**delete 10; delete 17; delete 50**

This sequence will cause nodes to merge because it removes nodes from the B-tree that are next to one another, which could lead the nodes to merge.

1. [1 mark] Suggest a sequence of four *insert* operations that will not result in changing the structure (number of nodes) of the B-tree.

Insert 4

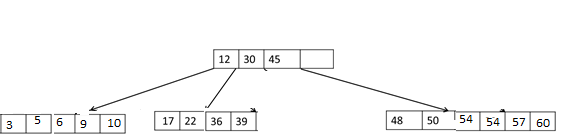
Insert 9

Insert 11

Insert 15

These procedures will add new values to existing nodes without splitting or merging them to retain the B-tree's structure.

1. [2 marks] Redraw the B-tree after executing the following sequence of operations:   
   insert 9; insert 60; insert 54



**Question 4:** [9 marks] Query Processing and Optimization

[Outcome D]

1. Using a COMPANY database, perform the following steps of heuristic query optimization for the query given below:

SELECT e.Fname, e.Ssn, p.Pname

FROM PROJECT p, DEPARTMENT d, EMPLOYEE e

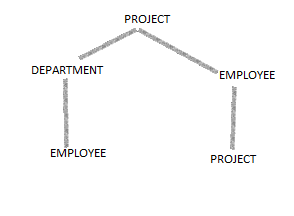
WHERE p.Dnum = d.Dnumber AND d.Dnumber = e.Dno AND

e.Salary < 20000 AND d.Dname = 'Procurement'

1. [1 mark] Translate the SQL query into a Relational Algebra query.

|  |
| --- |
| SELECT e.Fname, e.Ssn, p.Pname  FROM PROJECT p, DEPARTMENT d, EMPLOYEE e  WHERE p.Dnum = d.Dnumber AND d.Dnumber = e.Dno AND e.Salary < 20000 AND d.Dname = 'Procurement' |

1. [1 mark] Create an initial (canonical) Query Tree.



1. [1 mark] Move SELECT operations down the query tree, if applicable.

There are already SELECT operation at root, so no more required.

1. [1 mark] Apply the more restrictive SELECT operations first, if applicable.

|  |
| --- |
| SELECT  π Fname, Ssn, Pname  (  PROJECT  ×  DEPARTMENT  ×  EMPLOYEE  WHERE  p.Dnum = d.Dnumber  AND d.Dnumber = e.Dno  AND e.Salary < 20000  AND d.Dname = 'Procurement'  ) |

1. [1 mark] Replace CARTESIAN PRODUCT and SELECT with JOIN operations, if applicable.

|  |
| --- |
| SELECT  π Fname, Ssn, Pname  (  σ Salary < 20000  (  σ Dname = 'Procurement'  (  PROJECT p  JOIN DEPARTMENT d ON p.Dnum = d.Dnumber  JOIN EMPLOYEE e ON d.Dnumber = e.Dno  )  )  ) |

1. [2 marks] Move PROJECT operations down the query tree, if applicable.

|  |
| --- |
| SELECT  π Fname, Ssn, Pname  (  σ Salary < 20000  (  σ Dname = 'Procurement'  (  JOIN DEPARTMENT d ON p.Dnum = d.Dnumber  JOIN EMPLOYEE e ON d.Dnumber = e.Dno  WHERE  p.Dnum = d.Dnumber  AND d.Dnumber = e.Dno  )  )  ) |

1. [1 mark] Suggest an algorithm for implementing the SELECT operation of:

σ Dnum > 7 (PROJECT)

If the PROJECT relation has only one index on the Pnumber attribute

1. [1 mark] Suggest an algorithm for implementing the SELECT operation of:

σ Dnum = 7 AND Pnumber < 100 (PROJECT)

If the PROJECT relation has a clustering index on the Dnum attribute and a Primary index on Pnumber.

**Algorithm:**

A PROJECT relation with a single Pnumber index could implement the SELECT operation as follows:

|  |
| --- |
| First, index the Filter.  Pnumber values can be found by reading the index.  P-values below 7 must be removed from the output.  The next step: Get these documents  Apply the first-step filtered Pnumber values to the PROJECT relation.  Get data records for Dnum > 7. |

**Question 5:** Transactions

[Outcome E]

[4 marks] Answer the following questions based on schedule S1 that consists of two transactions:

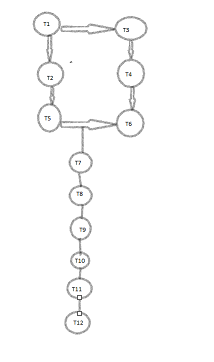
|  |  |
| --- | --- |
| **T1** | **T2** |
|  | R(a) |
|  | a = a + 10 |
| R(a) |  |
| a = a - 6 |  |
| R(b) |  |
|  | R(c) |
|  | W(a) |
|  | c = c - 2 |
| W(a) |  |
| b = b +15 |  |
| W(b) |  |
|  | W(c) |

1. [2 marks] For S1, assume that the initial values for a=60, b=80 and c=30. What would be the values of a, b and c after the S1 finishes.

The conflict serializability check ensures that no two schedule transactions access the same data item in a conflicting way.   
The recoverability check ensures that no transaction can change data while being processed, giving you piece of mind. This precaution prevents dirty reads, which occur when one transaction accesses data that another has not committed. Prevention makes a dirty read feasible.

1. [2 marks] S1 suffers from one of the concurrency control problems. Name the problem and explain its effect.

S2 is serialized because there are no cycles in it.

****

**Question 6:** Transactions

[Outcome E]

[6 marks] Answer the following questions based on schedule S2 (given below) that consists of three transactions:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **T1** | **T2** | **T3** |
| t=1 |  |  | R(d) |
| t=2 | R(a) |  |  |
| t=3 | R(b) |  |  |
| t=4 |  |  | R(b) |
| t=5 |  | R(c) |  |
| t=6 | W(b) |  |  |
| t=7 |  |  | R(a) |
| t=8 |  | R(a) |  |
| t=9 |  | W(c) |  |
| t=10 |  |  | R(c) |
| t=11 |  |  | W(a) |
| t=12 | W(a) |  |  |
| t=13 |  |  |  |

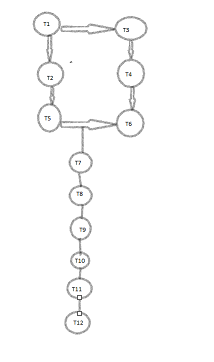
1. [2 marks] When a transaction advances to a *Partially Committed* state, mention two checks that take place before the transaction advances to the following state.

Because scheduling transactions cannot read/write the same data item at the same time, the conflict serializability check prevents this.

Uncommitted transactions cannot access successfully altered data due to the recoverability check. This prevents dirty reads, which occur when one transaction accesses uncommitted data.

1. [2 marks] Draw the serializability (precedence) graph of S2 and indicate whether S2 is serializable or not.

It has no cycles so it is serialized.

****

1. [2 marks] If each of T2 commits at t=13 and T3 commit at t=14 and then T1 aborts at t=15, indicate whether any transaction(s) need to be rolled back.

If T1 aborts at time t=15, there is no need to roll back any transactions because both T2 and T3 have already committed, and the changes that were made by T1 that aborted will not have any impact on the transactions that have already been committed. This is the case regardless of whether T2 commits at time t=13 or T3 commits at time t=14. Therefore, there is no need to perform a rollback.

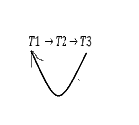
**Question 7:** Concurrency Control and Recovery

[Outcome F]

[4 marks] Given the transactions schedule S3 shown below, answer the following questions.

|  |  |  |
| --- | --- | --- |
| **T1** | **T2** | **T3** |
| Read\_lock(x) |  |  |
| Read\_item(x) |  |  |
|  | Read\_lock(y) |  |
|  | Read\_item(y) |  |
|  |  | Write\_lock(y) |
|  |  | Read\_lock(z) |
| Write\_lock(z) |  |  |
| Write\_lock(x) |  |  |
|  | Read\_lock(x) |  |

1. [2 marks] Draw the **wait-for** graph for the above schedule and explain the existing deadlock.



1. [1 mark] How can the deadlock in S3 be avoided?

It is possible to reduce the likelihood of deadlocks occurring by incorporating a deadlock detection and recovery method into S3. In order to accomplish this, it is necessary to perform routine checks for deadlocks and, if any are discovered, to cancel a transaction or transactions to try to eliminate the deadlock.

1. [1 mark] Which two-phase protocol prevents deadlocks? What is the main limitation of this protocol?

Strict Two-Phase Locking (S2PL) assures no deadlocks because it is a two-phase protocol. This protocol prevents transactions from locking up many resources for too long. After completing or aborting a transaction, it releases its locks.

**Question 8:** Concurrency Control and Recovery

[Outcome F]

[4 marks] Given the transactions schedule S4 shown below, answer the following questions.

|  |  |
| --- | --- |
| **T1** | **T2** |
| Read\_lock(y) |  |
| Write\_lock(x) |  |
| Read\_item(y) |  |
|  | Read\_lock(y) |
| Read\_item(x) |  |
| Unlock(y) |  |
| x = x - y |  |
| Unlock(x) |  |
|  | Write\_lock(x) |
|  | Read\_item(y) |
|  | Read\_item(x) |
|  | Unlock(y) |
|  | x = x \* y |
|  | Unlock(x) |

1. [2 marks] Does S4 follow the two-phase locking protocol? Explain.

The S4 schedule violates two-phase locking. Each transaction must commit or cancel before the protocol releases locks. S4's two-phase locking mechanism is violated since this transaction releases y's lock before writing to x.

1. [1 mark] Which 2PL technique (Basic, Conservative, Strict, and Rigorous) does S4 follow?

Schedule S4 implements the cautious Two-Phase Locking (2PL) approach. This approach enables a transaction to relinquish a lock on a resource prior to obtaining a lock on another resource, provided that it does not release a lock on a resource that is currently being used by a different transaction.

1. [1 mark] Assume that x=20 and y=5, what are the values of x and y after executing S4.

Final values:

x=75 and y=5