

Student ID	Name

For Instructor/TA only,

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total

Academic Honor Pledge

I affirm that I will not at any time be involved with cheating or plagiarism while enrolled as a student at Sungkyunkwan University.

I understand that violation of this code will result in penalties as severe as indefinite suspension from the university.

Your signature: _____

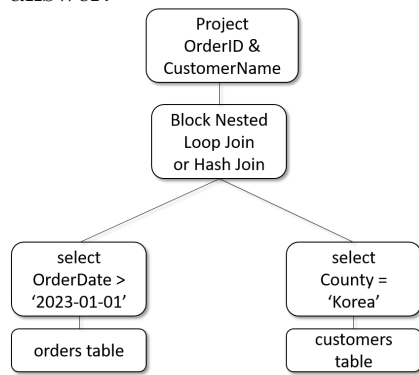
1. [Query Optimization (15 pts)] Suppose you are have **orders** (with 1,000,000 rows) and **customers** tables (with 100,000 rows). Each disk page holds maximum 1,000 orders records or 1,000 customers records. Consider the following query.

```
SELECT orders.OrderID, customers.CustomerName
FROM orders JOIN customers ON orders.CustomerID = customers.CustomerID
WHERE orders.OrderDate > '2023-01-01' AND customers.Country = 'Korea';
```

Suppose the orders are sorted by date in the table and 10% of the orders were made since 2023-01-01. Additionally, 20% of the customers live in Korea.

- (a) Draw an expression tree for a query execution plan that you think the most efficient. Describe how a cost-based optimizer would evaluate and choose between different query execution plans: If necessary, you may make specific assumptions on the database.

answer:



The join attribute is not sorted and there is no index, so a hash join or block nested loop join should be used. Although the select operation is shown as a separate node in the tree, it can be processed together with the join.

- (b) What is the expected number of block transfers if we employ Block Nested-Loop Join?

answer:

the number of blocks for orders: 1,000 and the number of blocks for customers: 100
Therefore, $100,000 + 1,000 = 101,000$ or $100,000 + 100 = 100,100$

If you considered the filtering, i.e., ($\rightarrow 10\% \times 1000 = 100$) and ($\rightarrow 20\% \times 100 = 2$)
the answer will be 300 or 202. But this is not a 100% correct answer. So, let us deduct 1 point.

- (c) What is the expected number of block transfers if we employ Hash Join? Recursive partitioning is not required, but the build input does not fit in memory.

answer:

Partitioning requires $(1,000 + 100)$ block reads and $(100 + 2)$ writes.
Build and probe phase requires $100 + 2$ reads.
Therefore, 1,304 block transfers are required.

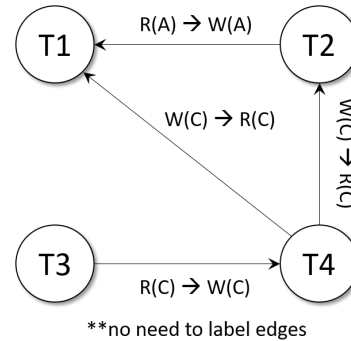
If you did not consider the filtering, the answer will be 3,300. But, again, this is not a 100% correct answer. So, let us deduct 1 point.

2. [Serializability (10pts)] Consider the following schedules where time increases from top to bottom.

time	T1	T2	T3	T4
1	R(A)			
2		R(A)		
3				R(C)
4			R(C)	
5			R(B)	
6				W(C)
7	W(A)			
8	R(C)			
9		R(C)		
10			W(B)	

Table 1: Schedule 1

(a) Draw a precedence graph:



(b) Which of the following serial schedules are conflict equivalent to the schedule above?

[1] T4, T2, T1, T3

[2] T3, T2, T4, T1

[3] T3, T4, T2, T1

[4] T1, T2, T4, T3

[5] T1, T4, T2, T3

[6] None of the above

answer: 3

3. [2PL (10 pts)] Suppose T1 is older than T2, and T2 is older than T3. The oldest transaction has the highest priority. No locks are released in the time frame shown. Note that not all requests may be granted immediately; in some cases, the transaction remains blocked, waiting for a preceding request to be completed.

time	T1	T2	T3
1	X(D)		
2		X(A)	
3	S(A)		
4		S(B)	
5			S(B)
6	S(C)		
7			X(B)
8		S(D)	
9			S(C)

The above schedule could not successfully occur whether one uses deadlock avoidance or not. Suppose we are not using any deadlock avoidance algorithms by default.

(a) Which lock requests are denied and blocked?

answer:

3, 7, and 8

(b) In which time step, deadlock will occur?

answer:

8

(c) Which transactions will deadlock?

answer:

T1 and T2

(d) If we were using the wound-wait deadlock avoidance algorithm, what would happen at time step 3?

- [1] The transaction requesting the lock would wait
- [2] The transaction requesting the lock would abort
- [3] The transaction that holds the lock would abort

answer:

3. T1 is requesting the lock T2 holds and it has higher priority, so T2 would abort.

4. [Timestamp-ordering (10pts)] Consider the following schedule under timestamp-ordering protocol. Initially, the write and read timestamps of data A, B, and C are all 0, and the timestamps of T1, T2, and T3 are 1, 2, and 3, respectively.

time	T1	T2
1	R(A)	
2		R(A)
3	W(B)	
4		W(B)
5	R(B)	
6		W(A)

Table 2: Schedule 3

time	T1	T2	T3
1	R(A)		
2		R(A)	
3			R(A)
4		R(B)	
5	R(C)		
6	W(A)		
7		R(A)	

Table 3: Schedule 4

- (a) Does Schedule 3 abort any transaction under timestamp-ordering protocol? Justify your answer.

answer: Yes. It aborts because R(B) by T1 will be rejected.

time	T1	T2	R-ts(A)	W-ts(A)	R-ts(B)	W-ts(B)
1	R(A)		1	0	0	0
2		R(A)	2	0	0	0
3	W(B)		2	0	0	1
4		W(B)	2	0	0	2
5	R(B)		2	0	1(Reject)	2

- (b) Does Schedule 4 abort any transaction under timestamp-ordering protocol? Justify your answer.

answer:

Yes. W(A) by T1 will be rejected.

	T1	T2	T3	R-ts(A)	W-ts(A)	R-ts(B)	W-ts(B)	R-ts(C)	W-ts(C)
1	R(A)			1	0	0	0	0	0
2		R(A)		2	0	0	0	0	0
3			R(A)	3	0	0	0	0	0
4		R(B)		3	0	2	0	0	0
5	R(C)			3	0	2	0	1	0
6	W(A)			3	1(Reject)	2	0	1	0

5. [Multi-version Timestamp Ordering (10 pts)] Consider the following schedule under multi-version timestamp-ordering (MVTSO) protocol. Initially, the write and read timestamps of data A, B, and C are all 0, and the timestamps of T1, T2, and T3 are 1, 2, and 3, respectively.

time	T1	T2
1	R(A)	
2		R(A)
3	W(B)	
4		W(B)
5	R(B)	
6		W(A)

Table 4: Schedule 5

time	T1	T2	T3
1	R(A)		
2		R(A)	
3			R(A)
4		R(B)	
5	R(C)		
6	W(A)		
7		R(A)	

Table 5: Schedule 6

- (a) Does Schedule 5 abort any transaction under MVTSO protocol? Justify your answer.

answer:

No, T1 does not abort because R(B) can access the old version of B. No need to draw the

table below.

time	T1	T2	R-ts(A)	W-ts(A)	R-ts(B)	W-ts(B)
1	R(A)		1	0	0	0
2		R(A)	2	0	0	0
3	W(B)		2	0	0	1
4		W(B)	2	0	0	2
5	R(B)		2	0	1(Read old ver.)	2

- (b) Does Schedule 6 abort any transaction under MVTSO protocol? Justify your answer.

answer:

Yes, T1 can not create a version because later transactions have already read the old version. Again, no need to draw the table below.

	T1	T2	T3	R-ts(A)	W-ts(A)	R-ts(B)	W-ts(B)	R-ts(C)	W-ts(C)
1	R(A)			1	0	0	0	0	0
2		R(A)		2	0	0	0	0	0
3			R(A)	3	0	0	0	0	0
4		R(B)		3	0	2	0	0	0
5	R(C)			3	0	2	0	1	0
6	W(A)			3	1(Reject)	2	0	1	0

6. [Recovery (10pts)] The following table shows a log file after a crash occurred. The DBMS employs the *immediate database modification*, which allows updates of an uncommitted transaction to be made to the buffer, or the disk itself, before the transaction commits

LSN	LOGS
1	<START T1>
2	<T1, X, 0, 5>
3	<START T2>
4	<T1, Y, 0, 7>
5	<T2, X, 5, 9>
6	<START T3>
7	<T3, Z, 0, 11>
8	<COMMIT T1>
9	<CHECKPOINT (T2,T3)>
10	<T2, X, 9, 13>
11	<T3, Y, 7, 15>
*C*R*A*S*H*	

Figure 1: Schedule 3

(a) Show below the log records appended by the recovery manager during recovery.

answer:

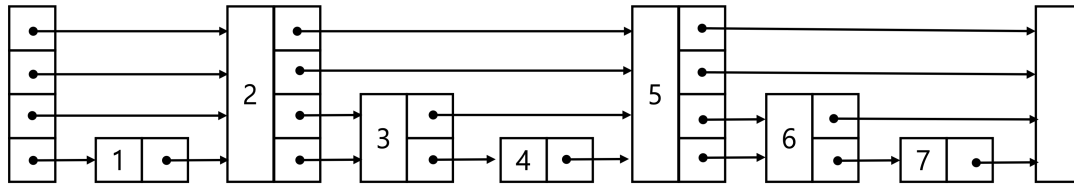
12 <T3, Y, 7>
13 <T2, X, 9>
14 <T3, Z, 0>
15 <T3, abort>
16 <T2, X, 5>
17 <T2, abort>

(b) What is the value of X at the end of the recovery?

answer:

5

7. [Skip Lists (10 pts)] Consider the following Skip Lists.



(a) How many key comparisons are required to search 4?

answer:

6 times.

[top level]

visit node 2 and compare(2,4)

visit node 5 and compare(5,4). query key is smaller. So, backtrack

[3rd level]

visit node 5 and compare(5,4). query key is smaller. So, backtrack

[2nd level]

visit node 3 and compare(3,4).

visit node 5 and compare(5,4). query key is smaller. So, backtrack

[bottom level]

visit node 4 and compare(3,4). FOUND

(b) How many pointers need to be updated to delete 2?

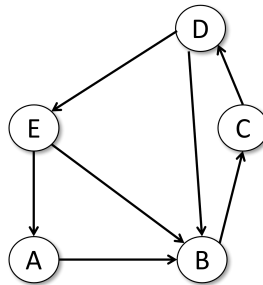
answer:

4. One pointer per level needs to be updated.

8. [NoSQL (10 pts)] For each of the following statements, indicate whether it is TRUE or FALSE. You will get 2 points for each correct answer, -2 points for each incorrect answer, and 0 point for each answer left blank or both answers marked.

- | | T | F |
|---|--------------------------|--------------------------|
| (a) One of the goals of the MapReduce framework is to avoid using low-level programming techniques in developing large-scale distributed and parallel applications. T | <input type="checkbox"/> | <input type="checkbox"/> |
| (b) In SkipLists, write operations are generally faster than read operations. F | <input type="checkbox"/> | <input type="checkbox"/> |
| (c) The primary goal of compaction in LSM-trees is to improve search performance. T | <input type="checkbox"/> | <input type="checkbox"/> |
| (d) The stall issue in an LSM tree arises from the accumulation of read queries. F | <input type="checkbox"/> | <input type="checkbox"/> |
| (e) MongoDB supports all the functionalities of a relational DBMS except for DDL (Data Definition Language). F | <input type="checkbox"/> | <input type="checkbox"/> |

9. [Graph Processing (10 pts)] Consider the following graph. Determine the nodes with the highest and lowest rank after running the PageRank algorithm on the network. Using the chromatic engine, how many steps are needed per iteration to maximize parallelism in this graph? Briefly justify your answer for each question.



highest rank:

B has 3 incoming edges.

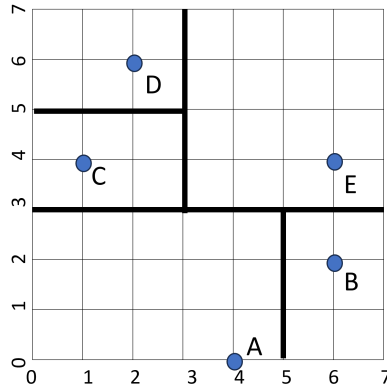
lowest rank:

A has the lowest rank because, except for C, all nodes have only one incoming edge. Due to the characteristic of page rank where the rank decreases as the distance from the highest-ranked node, C, increases, A, being the farthest, has the lowest rank.

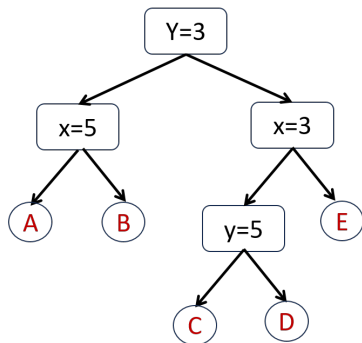
The number of steps required per iteration:

The answer is 3. First, let's color B, which has the highest rank, in RED. The neighboring C cannot be RED, so let's color it BLUE. D cannot be either RED or BLUE, so let's color it YELLOW. E has no edge with C, so it can be colored BLUE. A is not connected to D, so it can be colored YELLOW. Therefore, all nodes can be colored with three colors, and nodes with the same color can perform their operations simultaneously.

10. [Spatial Indexing (10 pts)] Consider the following spatial data and K-d-tree.



(a) Fill in the labels (i.e., A, B, C, D, E) for the leaf nodes within the circles in the following tree structure.



(b) How many nodes need to be visited to process the range query $1 < X < 4$ and $2 < Y < 6$? If the same node is visited multiple times, count it only once.

answer:

8. All nodes except node B are visited by following the red path.

