Database Systems: Mid Exam

Date: 18th September 2019 **Duration: 1.5 hrs**

1. No clarifications during the exam.
2. Make *reasonable assumptions* and *clearly state* them to answer *ambiguous* questions.
3. Show your steps. Be concise and organized.
4. Calculators allowed. Sharing of calculators *not* allowed.

34
21
13

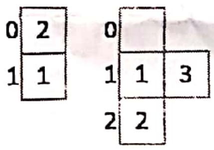
9 7
1 5

1) Construct a B+-tree for the following set of values (2, 3, 5, 7, 11, 17, 19, 23, 29, 31)
Assume that the tree is initially empty and values are inserted in ascending order.
(A) Construct the B+-tree if the number *m* of pointers that will fit a node is 4.
(B) Show the form of the B+-tree after each operation of the sequence:
Insert 9; Insert 10; Insert 8; Insert 6; Insert 1 **[10]**

2) Show the state of a linear hashing file after each insert of keys in the sequence: 1, 2, 3, 5, 8, 13, 21, 34. Assume bucket capacity = 1 and family of hash functions = mod 2, mod 4, mod 8, mod 16, ... The file has no directory but may have an overflow area. For e.g. the following diagram shows the state after inserting keys 1, 2, 3. **[10]**

2
2

Next bucket to split	0	1						
H(i, k)	mod(k,2)	mod(k,2)						
H(i+1, k)		mod(k,4)						



8
1
2 34
3
5 13 21

3) Suppose we place the following data in a grid file with dimensions for speed and ram only. The partitions are at speeds of 310, 375 and 425, and at ram of 40 and 75. Suppose also that only two points can fit in one bucket. Suggest good splits if we insert points at: (a) Speed=250 and ram=48; (b) Speed=333 and ram=48. **[10]**

Model	*Speed*	*Ram*	*Hard-disk*
A	300	32	6.0
B	333	64	4.0
C	400	64	12.7
D	350	32	10.8
E	450	96	14.0
F	400	128	12.7
G	450	128	18.1
H	233	32	4.0
I	266	64	6.0
J	300	64	6.0
K	350	64	12.0
L	400	128	6.0

333 18

333 64
333 48

4) Give examples to show that:

(a) Projection cannot be pushed below set union.

(b) Projection cannot be pushed below set or bag difference. [10]

5) (a) Using the statistics below, estimate the sizes of the results of the following expressions.

W(a,b)	X(b,c)	Y(c,d)	Z(d,e)
$T(W) = 100$	$T(X) = 200$	$T(Y) = 300$	$T(Z) = 400$
$V(W, a) = 20$	$V(X, b) = 50$	$V(Y, c) = 50$	$V(Z, d) = 40$
$V(W, b) = 60$	$V(X, c) = 100$	$V(Y, d) = 50$	$V(Z, e) = 100$

(a) $W \bowtie X \bowtie Y \bowtie Z$ 8000

(b) $\sigma_{a=10}(W)$ 5

[10]

123 423

223

Scanned by CamScanner

In all questions, show your reasoning and steps, concisely.

1. Given T(R1)=50,000, T(R2)=30,000, T(R3)=20,000, S(R1)=S(R2)=S(R3)=1/10 block,
Memory=101 blocks. Among the variations of iteration and merge
joins, what will be a cost-efficient join order and strategy to
compute R1 JOIN R2 JOIN R3, if:

(a) The relations are not contiguous
(b) The relations are contiguous

2. Consider the following UNDO log sequence:

<START T1> <T1 X 5> <START T2> <T1 Y 7> <T2 X 9> <START T3> <T3 Z 11>
<COMMIT T1> <START CKPT(T2,T3)> <T2 X 13> <T3 Y 15> *C*R*A*S*H*

(a) How far back should the recovery manager read the log ?
(b) List the actions of the recovery manager during recovery.
(c) What is the value of X at the end of the recovery ?

3. Consider the following two transactions and schedule:

T0: R0(A) W0(A) R0(B) W0(B) C0

T1: R1(A) R1(B) C1

(a) Show that this schedule is not conflict serializable.
(b) Show how 2PL can ensure a serializable schedule for T0 and T1.

4. For each of the schedules:
(a) R1(A) R2(B) R3(C) R1(B) R2(C) R3(D) W1(A) W2(B) W3(C)
(b) R1(A) R2(B) R3(C) R1(B) R2(C) R3(A) W1(A) W2(B) W3(C)
Insert shared, exclusive and update locks, and unlock actions.

Database Systems

End Semester Exam

Date: 16th November 2019 **Duration: 3 hrs; Marks: 90**

1. **No clarifications during the exam.**
2. **Make *reasonable assumptions* and *clearly state* them to answer *ambiguous* questions.**
3. **Show steps. Be concise and organized.**
4. **Calculators allowed. Sharing of calculators *not* allowed.**

1) (a) What are the major components of disk-access time, to read/write a block from disk? seek Rotation Transfer.
(b) Which of these components depends on block-size? [4]

2) Answer true (if sensible) or false:
(A) An unsorted file can have a dense first-level index. ✓
(B) An unsorted file can have a sparse first-level index. ×
(C) A sorted (sequential) file can have a dense first-level index.
(D) A sorted file can have a sparse first-level index. ✓
(E) We can build a dense, 2nd level index for a dense index. × ×
(F) We can build a dense, 2nd level index for a sparse index. × [6]

3) What is the worst-case complexity of access (read/insert/delete) in conventional (sparse / dense) primary indices? Explain how. [5]

4) Suppose blocks can hold either 10 records or 99 keys and 100 pointers. Suppose the average B+ tree ~~B-tree~~ node is 70% full (69 keys + 70 pointers). The data file is a sequential file, sorted on the search key, with 10 records per block. The B-tree is a dense index. Determine: (A) Number of blocks needed for a 1 million record file. (B) Average number of disk I/Os to retrieve a record given its search key. Assume nothing is in memory initially, and the search key is the primary key. [10] 110247 5.

Doubt ✓

5) Suppose B(R) = B(S) = 10,000. What value of M would we need to compute R ⋈ S using nested loop join with no more than 1 lakh disk I/Os? 1002 [10]

6) Show that we can always push a projection below both branches of a bag union. [15]

7) Using the statistics below for 4 relations, estimate the size of their join. [10] 12 ?

E(a,b,c)	F(a,b,d)	G(a,c,d)	H(b,c,d)
T(E) = 1000	T(F) = 2000	T(G) = 3000	T(H) = 4000
V(E, a) = 1000	V(F, a) = 50	V(G, a) = 50	V(H, b) = 40
V(E, b) = 50	V(F, b) = 100	V(G, c) = 300	V(H, c) = 100
V(E, c) = 20	V(F,d) = 200	V(G, d) = 500	V(H, d) = 400

8) A consistency constraint is $0 \leq A \leq B$. For the transaction: A := A+B; B := A+B; (i) Tell if the transaction preserves consistency, (ii) Add read and write actions and show the effect of the steps on main-memory and disk, (iii) Show the undo-log records, redo-log records and undo-redo log records, assuming initially A=5 and B=10. [15]

9) What happens when the following sequence is processed by a validation-based scheduler:
$R_1(A,B)$; $R_2(B,C)$; V_1; $R_3(C,D)$; V_3; $W_1(A)$; V_2; $W_2(A)$; $W_3(B)$; [5]

10) Consider an object hierarchy with relation R1 at the top-level containing 4 tables t1, t2, t3, t4. Table t2 contains 2 records f2.1 and f2.2. Table t3 contains 2 records f3.1 and f3.2. (i) If a transaction T1 contains IS lock at R1 and S lock at t2, what locks will a transaction T2 get to access f3.1 in X mode, if it can? (ii) If T1 contains IX lock at R1 and X lock at t2, what locks will T2 get to access f2.2 in X mode, if it can? [10]