This question paper contains 6 printed pages.

3237

Your Roll No. ....

MECTA **J**

COMPUTER TECHNOLOGY AND APPLICATIONS

Paper – CS . 653 (*i*)

(Distributed DBMS)

Time : 3 hours *Maximum Marks* : 100

(Write your Roll No. on the top immediately on receipt of this question paper.)

Attempt any five questions out of 8 questions. All questions carry equal marks.

1.1 Consider the following Partitioning techniques used in parallel databases **(6)**

a) Round Robin Partitioning
b) Hash Partitioning
c) Range Partitioning

How the following type of queries are handled in each of above type of partitioning

- Point queries
- Range queries

1.2 Consider the following partitioning using virtual processors **(4)**

No of processors = 6 <0,1,2,3,4,5>
No of vitual Processors = 12 <0,1,…,11>
Partitioning Vector <10,20,…..110>
In which processor(s) the tuples with key values lie?

a. 15-16
b. 75-85
c. 45-75
d. 120-200

1.3 Consider the following processors with key data values **(10)**

P0	P1	P2	P3
32	33	69	78
87	44	25	23
77	66	55	44
99	89	25	65

Turn over

Using partitioning vector <40,60,80>

(1) Show content of each processor in **Range Partitioning Sort** after each step of processing.

(2) Show content of each processor in **Parallel external sort-merge** after each step of processing.

2.1 Consider the following processors with tuples **(10)**

P0	P1
R relation	**S relation**
<10,A1>	<15,red>
<15,A3>	<10,blue>
<22,A4>	<35,white>
<35,A5>	<15,yellow>
<22,A6>	<35,cream>
<35,A7>	<22,green>

R relation is in P0 and S relation is in P1 processor

Do a **parallel join** on the first field of R & S using partitioning vector <20>

Show content of each processor after each step of processing.

2.2 Explain the following parallel operations

a) Parallel equi-Join using hash partitioning **(5)**

b) Asymmetric Fragment and Replicate Join **(5)**

3.1. (a) What is Write ahead log?

(b) In WAL when are dirty pages written to Database?

(c) What is done during checkpoint ?

(d) When is log block written to log file ? **(8)**

3.2 For transactions *T0* and *T1* (*T0* executes before *T1*): **(4)**

T0: **read** (*A*)
A := *A* - *50*
Write (*A*)
read (*B*)
B := *B* + *50*
write (*B*)

T1 : **read** (*C*)
C := *C* - *100*
write (*C*)

The following Log file is found after a system crash in a database using **Immediate database modification** scheme

<T_0 start>	<T_0 start>	<T_0 start>
<T_0, *A*, 1000, 950>	<T_0, *A*, 1000, 950>	<T_0, *A*, 1000, 950>
<T_0, *B*, 2000, 2050>	<T_0, *B*, 2000, 2050>	<T_0, *B*, 2000, 2050>
	<T_0 commit>	<T_0 commit>
	<T_1 start>	<T_1 start>
	<T_1, *C*, 700, 600>	<T_1, *C*, 700, 600>
		<T_1 commit>
(a)	(b)	(c)

What action needs to be taken in each of above cases and why?

3.3 The following information is found in the log file after a crash **(8)**

<*T0* **start**>
<*T0*, *A*, 0, 100>
<*T0* **commit**>
<*T1* **start**>
<*T1*, *B*, 0, 100>
<*T2* **start**>
<*T2*, *C*, 0, 100>
<*T2*, *C*, 100, 200>
<*T3* **start**>
<checkpoint {*T1*, *T2*,T3}>
<*T4* **start**>
<*T3*, *A*, 100, 300>
<*T1* **commit**>
<*T3*, *D*, 0, 100>

How the above log file is processed to recover the database.?

4.1 In majority protocol for replication updation. **(10)**

(a) How many locks are acquired for read and write operations?

(b) While reading which copy is read?

(c) While writing, how many copies are updated?

(d) For a system with 10 replicated sites, maximum of how many sites can be down without affecting the successful operation of the majority protocol?

(e) Why nothing needs to be done when a failed copy site recovers?

4.3 In quorum consensus protocol for replication updation (6)

(a) What constraint Read and Write Quorum should satisfy?

(b) How can you reduce Read locking overhead and make it faster?

(c) When will you use this protocol instead of majority protocol?

4.4 (a) In primary copy protocol, how many copies needs to be updated? (4)

(b) What happens when one of the copy site is down?

5. In Two-phase commit protocol used in a distributed database, (20)

(a) What are the two phases ?

(b) What happens when coordinator does not receive some readyT message from one site?

(c) What will a participating site do when it does not receive any Commit T/ Abort T message from coordinator after it sent its ready T message?

(d) A participating site recovers from failure and finds ready T in its log file. What action will it take?

(e) A coordinator site recovers from failure and finds ready T in its log file. What action will it take

(f) When is a Participating site blocked?

(g) Is there any condition under which coordinator can be blocked?

(h) What initial processing participating site has to do after recovery from failure?

6.1 Describe in brief Three-Phase Commit (3PC) protocol for distributed transaction **(10)**

6.2 How is coordinator failure handled in 3PC **(4)**

6.3 What happens when the new coordinator also fails? **(4)**

6.4 What is the advantage of 3PC over 2PC **(2)**

7.1 Write the query transformation rules for the following **(6)**

a) Commuting selection with binary operations (Cross Product, Join, Union)

b) Commuting projection with binary operations (Cross Product, Join, Union)

7.2 Map the following query into an operator tree and transform it into an Optimized tree using the restructuring algorithm. **(8)**

```
SELECT  ENAME, PNAME,DUR
FROM    PROJ, ASG, EMP
WHERE   ASG.ENO = EMP.ENO
AND     ASG.PNO = PROJ.PNO
AND     PROJ.PNAME = "CAD/CAM"
AND     (DUR > 20)
AND     (TITLE = "Manager")
AND     (BUDGET > 50000)
```

Assume the following relational schema

```
EMP     (ENO,   ENAME,   TITLE)
PROJ    (PNO,   PNAME,   BUDGET, LOC)
ASG     (ENO,   PNO,     RESP,   DUR)
```

7.3 Assume that the relation PROJ in above schema is horizontally fragmented in **(6)**

PROJ1 = $\sigma_{PNO <= \text{"P5"}}$ (PROJ)
PROJ2 = $\sigma_{PNO > \text{"P5"}}$ (PROJ)

Transform the following query into a reduced query on fragments

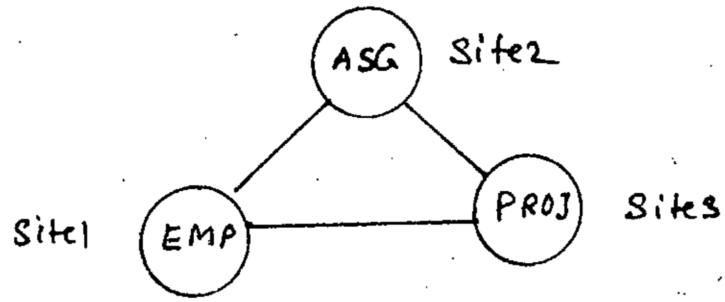
```
SELECT  BUDGET
FROM    PROJ, ASG
WHERE   PROJ.PNO = ASG.PNO
AND     ASG.PNO = "P5"
```

8.1 Consider Relation R & S on site 1 and 2 respectively. **(8)**
How will you perform join on attribute A using semi join approach?
Write the various processing steps.
Under what situation this would be less costly operation?

8.2 Consider the following query expressed in relational algebra (6)

PROJ $\bowtie$ PNO ASG $\bowtie$ ENO EMP

Whose join graph is given below.



Assume all sites are connected to each other and the following information

size (EMP)	= 400
size (ASG)	= 200
size (PROJ)	= 200
size (EMP $\bowtie$ ASG)	= 100
size (ASG $\bowtie$ PROJ)	= 400

Describe an optimal join program based on the objective function of total transmission time .

8.3 For the join graph and other information given in 8.2, describe an optimal join program that minimizes response time (consider only communication) (6)