MANAV RACHNA

MANAV RACHNA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE & TECHNOLOGY

"T3 Examination, Dec 2018"

Semester III
Subject: Database Management Systems
Branch: CSE
Course Type: Core
Time: 3 Hours
Max.Marks: 100

Date of Exam: 6/12/2018
Subject Code: CSH-205-T
Session: II
Course Nature: Hard
Program: B.Tech.
Signature: HOD/Associate HOD:

PART-A [10]

Q1. A file includes records with following hash-keys: - 5659, 1074, 1620, 3943, 9208, 2369, 3760, 4692, 4871, 1821, 7115. The file uses eight buckets – 0 to 7. Each bucket is one disk block and holds two records. Load these records into the file in the given order using hash function h(K) = k mod 8. Calculate the average number of block accesses for the same.

PART-B [10] each question carries 10 marks
Attempt any one

Q2. Using SQL perform (i)Create table employee (eno char(3), ename varchar(20), eloc varchar(20), salary number(5)) [4+2+2+2]

- PRIMARY KEY – eno
- ename cannot be left blank,default eloc is chennai

(ii) Find the employee names and their corresponding location for the employes having salary Between 30000 and 50000.
(iii)Find the employee names and their corresponding salary for the employees living in mumbai.
(iv)Create a view which is based on the details regarding the employee location and their corresponding employee names.

Q3. (a)You are given the following relational schema [2*5]

- lives(person-name,street,city)
- works(person-name, company-name,salary)
- located-in(company-name,city)
- manages(person-name,manager-name)

Write relational algebra to answer the following queries:

A. Find the name of all employees (i.e., persons) who work for the City Bank company (which is a specific company in the database).
B. Find the name and city of all employees who work for City Bank.
C. Find the name, street and city of all employees who work for City Bank and earn more than $10,000.
D. Find all employees who live in the same city as the company they work for.
E. Find all persons who do not work for City Bank.

PART-C [40] each question carries 20 marks
Attempt any two

Q4 a) Draw an ER diagram for the given scenario;

Suppose that you are designing a schema to record information about reality shows on TV. Your database needs to record the following information:

_ For each reality show, its name, genre, basic_info and participants name. Any reality show has at least two or more participants.

****

_ For each producer, the company name, company country. A show is produced by exactly one producer. And one producer produces exactly one show.
_ For each television, its name, start year, head office. A television may broadcasts multiple shows. Each show is broadcasted by exactly one television.
_ For each user, his/her username, password, and age. A user may rate multiple shows, and a show may be rated by multiple users. Each rating has a score of 0 to 10,
Draw an entity relationship diagram for this database. [14]

b) Convert the E-R diagram (4.a) into a set of relations [6]

Q5 a) Consider the relational scheme R (A,B,C,D,E,F) and FD A BC, C -» A, D -> E, F -> A, E -> D
Is the decomposition of R into R1 (A, C, D), R2 (B, C, D) and R3 (E, F, D) lossless? [5]

b) Consider the universal relation R = {A, B, C, D, E, F, G, H, I, J} and the set of functional dependencies F={ {A,B} -> {C},A} -> {D,E},{B}->{F},{F} ->{G,H},{D} -> {I,J}}.
i) What is the key for R? [5]
ii) Decompose R into 2NF, then 3NF relations. [10]

CEGHIJ

Q6 A relation R is defined as follows.
R = (name, street, city, state, postal_code). Here, name is unique, and for any given postal code, there is just one city and state.

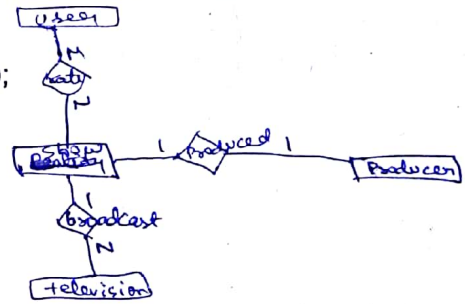
i) Give a set of FDs for this relation.
ii) What are the candidate keys?
iii) Is R in 3NF? 2NF? Explain why?
iv) If R is not in 3NF, normalize it into 3NF relations [4x5 = 20]

PART-D [40] each question carries 20 marks

Attempt any two

Q7. Consider the following two transactions: *T*1: read(*A*);

Read (B);
if A = 0then B := B + 1;
Write (B).
T2: read (B);
Read (A);
if B = 0then A := A + 1;
Write (A).



Let the consistency requirement be A = 0 ∨ B = 0, with A = B = 0 the initial values.
a. Show that every serial execution involving these two transactions preserves the consistency of the database.
b. Show a concurrent execution of T1 and T2 that produces a nonserializable schedule.
c. Is there a concurrent execution of T1 and T2 that produces a serializable schedule?

[4+8+8]

Q8. (a) Consider the following two transactions:

T31: read(A);
read(B);
if A = 0then B := B + 1;
write(B).
T32: read(B);
read(A);
if B = 0then A := A + 1;
write(A).

****

Add lock and unlock instructions to transactions T31 and T32, so that they observe the two-phase locking protocol. Can the execution of these transactions result in a deadlock? [5+5]

(b) Show that the first schedule is possible through time stamping but not through 2 phase locking, whereas the second schedule follows the reverse.

step	T_0	T_1	T_2
1	write(A)		
2		write(A)	
3			write(A)
4	write(B)		
5		write(B)	

step	T_0	T_1	Precedence remarks
1	lock-S(A)		
2	read(A)		
3		lock-X(B)	
4		write(B)	
5		unlock(B)	
6	lock-S(B)		
7	read(B)		$T_1 \rightarrow T_0$
8	unlock(A)		
9	unlock(B)		

[10]

Q 9. After a systems failure, the undo-redo recovery log has the following entries:

<START T1>
<T1 A 1 2>
<START T2>
<COMMIT T1>
<START T3>
<T3 A 2 3>
<START T4>
<CKPT(T2,T3,T4)>
<T2 B 10 20>
<COMMIT T2>
<START T5>
<T5 D 1000 2000>
<T4 C 100 200>
<COMMIT T5>
<START T6>
<END CKPT>
<T6 D 2000 3000>

An entry <T, X, u, v> means that transaction T has updated the value of X from u (the old value) to v (the new value). <CKPT(...)> denotes the beginning of a checkpoint and lists the currently active

****

transactions. <END CKPT> is written to disk once all dirty pages of the active transactions have been flushed to disk. The redo phase preceeds the undo phase during the recovery.

1. Which are the transactions whose actions the recovery manager needs to redo?
2. Which are the transactions whose actions the recovery manager needs to undo?
3. Indicate the actions of the recovery manager on all the elements, separately during the Redo and the Undo phase.

[5+5+10]

****