THE OPEN UNIVERSITY OF SRI LANKA
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

FINAL EXAMINATION 2014
BACHELOR OF SOFTWARE ENGINEERING – LEVEL 5

ECI5266 – ADVANCED DATABASE SYSTEMS

DATE: 03RD SEPTEMBER , 2014 **TIME: 0930-1230 HRS**

This question paper consists of eight questions in seven pages. Answer **FIVE** questions **ONLY.**

Question 1 – Database Programming (20 marks)

Consider the following relations in a bank database:

Branch (*bName*:char(15), *city*:varchar(12), *phone*:varchar(10))
Customer (*custno*:integer, *cName*:char(15), *gender*:char(1), *birthday*:date)
Account (*accno*:integer, *branchname*:char(15), *acctype*:char(1), *balance*:real)
AC (*accno*:integer, *custno*:integer)

The attributes of the Branch relation are the name of the branch (*bName*), *city*, and *phone*. The Customer relation has attributes to record the customer number (*custno*), name (*cName*), *gender* ('M' or 'F'), and *birthdate* of customers. The Account relation consists of account number (*accno*), branch name (*branchname*), account type (*acctype*) which may be individual (acctype='I') or joint (acctype='J'), and account balance (*balance*). The attribute *branchname* in Account relation is a foreign key that references Branch. An individual account belongs to a single customer, while a joint account is held by two or more customers. The attribute *accno* and *custno* attribute in AC relation reference the relations Account and Customer respectively. The primary keys of all relations are underlined.

(a) Express the following queries in SQL.

i. For each customer with a total balance exceeding 50,000 LKR in all his/her accounts together, display the customer number and total balance.

(3 marks)

ii. For each branch, display the branch name, account number and balance of each account that has a balance greater than twice the average balance of all accounts at that branch. In the resulting report, order the accounts of each branch in the descending order of account balance.

(5 marks)

(b) It is required to record the number of account holders for each account by adding an attribute named holders. The value of this attribute will be 1 for individual accounts and greater than 1 for joint accounts.

i. In the Account table, add a new attribute named *holders* of integer type with a default value of 1.
(1 mark)

ii. Write a T-SQL procedure to update the holders attribute (that has just been added) only for joint accounts.
(5 marks)

iii. Write a trigger to decrease the holders attribute value when a row is deleted from the table AC only if it is for a joint account. If the updated value of holders is one for the joint account, then change the acctype to 'I'.
(6 marks)

Question 2 – Object Relational Databases (20 marks)

Consider the following object relational database schema for a database of insurance policies for cars:

Object types:

Customer_t (cid: char(6), name: varchar(15), birthdate: date, phone: char(10), address: varchar(50))
Car_t (regno: char(9), make: varchar(12), model: varchar(10), mdate: date, owner: ref customer_t, value: number(8,2))
Claim_t (claimno: char(12), cdate: date, amount: number(8,2), claimant: ref customer)
Claim_ntab table of claim_t
Policy_t (pid: char(7), sdate: date, edate: date, inscar: ref car_t, premium: number(6,2), claims: claim_ntab)

Tables:

Customers of Customer _t (cid primary key)
Cars of car_t (regno primary key, owner references Customers)
Policies of policy_t (pid primary key, inscar references Cars)
Nested table claims store as claims_ntable

The tables named Customers, Cars, and Policies contain tuples for all customers, cars and policies respectively. Their attributes are indicated by the corresponding types.

Attributes of Customers are customer id (cid), name, date of birth, phone and address. Attributes of Cars are registration number (regno), make, model, date of manufacture, owner of the car, and insured value of the car.

Attributes of Policies policy id (pid), starting date, ending date, insured car, annual premium and the claims made.

Attributes of the nested table of claims are claim number (claimno), claim date (cdate), the amount of claim and the claimant.

The attribute types are specified in the type descriptions above, as also are the primary keys and referential constraints in the table schema.

(a) Write Oracle OBJECT SQL statements to answer the following queries (use columns of REF type instead of joins to link tables):

i. Find the average insurance premium on cars owned by customers aged between 20 and 25 years. Sysdate contains current date and the function month_between(d1,d2) gives the number of months in decimals between the dates d1 and d2 where d1>d2.

(3 marks)

ii. For each make and model of car, find the total claim amount on policies that expire between 1 Jan 2004 and 31 Dec 2004.

(4 marks)

(b) Write Oracle object SQL to insert a claim against an existing policy that has a pid of SL12354, given the following claim details: claim number: 001, claim date: 12 July 2004, claim amount: 2000, and customer id of claimant: S25431. Assume that the claimant is already present as a customer in the database.

(2marks)

(c) Write Oracle object SQL to calculate the renewal premium for a given policy in the database using the following logic:
If the policy had no claims or the total claim is less than 1000, then the new premium is the same as the current premium. If the total claim on a policy is greater than or equal to 1000, then the new premium is to be 20% more than the current.

(8 marks)

(d) Using the method defined above in (c), get the renewal premium for the policy of a car with registration number SLA984.

(3 marks)

Question 3 – Relational Algebra (20 marks)

Consider the following schema:

Suppliers (*sid*: integer, *sname*: string, *address*: string)
Catalog (*sid*: integer, *pid*: integer, *cost*: real)
Parts (*pid*: integer, *pname*: string, *color*: string)

Write **Relational Algebra** expressions for the following:

(a.) Find the *names* of suppliers who supply red parts.

(3 marks)

(b.) Find the *sids* of suppliers who supply red or green parts.

(3 marks)

(c.) Find the *sids* of suppliers who supply some red part and some green part that costs less than 100 dollars.

(4 marks)

(d.) Find the *sids* of suppliers that supply every red part.

(5 marks)

(e.) List the names of all suppliers and number of parts they are supplying. Number of parts should be printed as zero for suppliers who are not supplying any parts.

(5 marks)

Question 4 – Disk and Files (20 marks)

(a.) Explain the time spent in accessing a block in a disk.

(4 marks)

(b.) Consider a disk with the following characteristics: block size 512 bytes, number of blocks per track 50, number of tracks per surface 100, average seek time 10 msec. The disk consists of 4 double-sided platters. Suppose that a file containing 1000 records of 100 bytes each is to be stored on such a disk and that no record is allowed to span two blocks:

i. How many records of 100 bytes could be stored in a track?

(2 marks)

ii. How many tracks are needed to store the entire file?

(2 marks)

iii. If disk platter rotates at 5400 rpm (revolutions per minute), calculate transfer time of one track of data.

(2 marks)

iv. Assuming that the blocks in the file are stored sequentially and has only the data of the file in the disk, what is the time taken to read the file content?

(3 marks)

(c.) Briefly explain one technique used in RAID to improve performance and one technique used to improve reliability.

(3 marks)

(d.) Illustrate the two alternatives available for storing fixed length records in a page. List one advantage of each alternative over the other.

(4 marks)

Question 5 – Indexing and Query Processing (20 marks)

(a.) What are the steps in Query Processing? Explain each step.

(4 marks)

(b.) What is the justification for using I/O costs as the main measure in relational query optimizer?

(2 marks)

(c.) Briefly explain clustered indexes and un- clustered indexes.

(4 marks)

(d.) Consider the following schema:

```
Emp (enum:integer; ename:string, location: string)

Dept (dnum:integer,dname:string,manager:integer,budget:float)
```

Consider the following query:

```
SELECT      Emp.ename
FROM        Emp, Dept
WHERE       Emp.enum = Dept.dnum AND Dept.budget >= 100000
```

i. Write TWO relational algebra expressions for the above SQL query in which, natural join is used for one and Cartesian product for the other. Make sure to do selection before join tables.
(2 marks)

ii. Draw the graphical representation of the relational algebra expressions in (i.).
(2 marks)

iii. What query plan in (ii.) would you suggest to execute on a DBMS and briefly explain the query plan you consider?
(3 marks)

iv. What indexes would you suggest to speed up the query plan considered in (iii.)?
(3 marks)

Question 6 – Physical Database Design and Tuning (20 marks)

Consider the following relational schemas for recording information on suppliers, parts and shipments:

Suppliers (sno, sname, status, city)
Parts (pno, pname, weight, location)
Shipments (sno, pno, quantity)

Suppliers table consists of 100 pages with 10 tuples per page, Parts table has 300 pages of 10 tuples each, and Shipments has 1000 pages of 100 tuples each. There are 20 buffer pages.

Consider the query:

```
SELECT s.sname
FROM suppliers s, shipments sh
WHERE s.sno = sh.sno
and s.status >= 60
and sh.quantity > 500;
```

Assume that only 40% of Suppliers have a status of 60 or above, 10% of shipments have a quantity greater than 500. If there are no indexes on either Suppliers or Shipments:

(a.) What query plan would you suggest to execute on a DBMS and briefly explain the query plan you consider? Cost estimation is **not** essential.
(8 marks)

(b.) What would be the **cost** of a plan that performs selections before the join and uses a sort-merge join? Describe how you arrived at the cost step by step.

(12 marks)

Question 7 – Transactions and Concurrency Control (20 marks)

(a.) Briefly explain the properties of a transaction.

(4 marks)

(b.) Briefly explain what an *unrepeatable read* is.

(2 marks)

(c.) Briefly explain the rules in Strict 2 Phase Locking Protocol.

(3 marks)

(d.) Briefly explain what the *phantom problem* is.

(3 marks)

(e.) Consider the following sequence of actions, listed in the order they are submitted to the DBMS. The DBMS processes actions in the order shown. The **Strict 2PL** is used for concurrency control. If a transaction is blocked, assume that all of its actions are queued until it is resumed; the DBMS continues with the next action of an unblocked transaction.

T1:R(X), T2:W(X), T2:W(Y), T3:W(Y), T1:W(Y),T3W(X), T1: Commit, T2: Commit, T3: Commit

Assume that older transaction has higher priority always.

i. Draw a transaction schedule and briefly explain how the above three transactions are running till commit those without deadlocks in schedule. Follow *Wait-Die* policy to deal with deadlock.

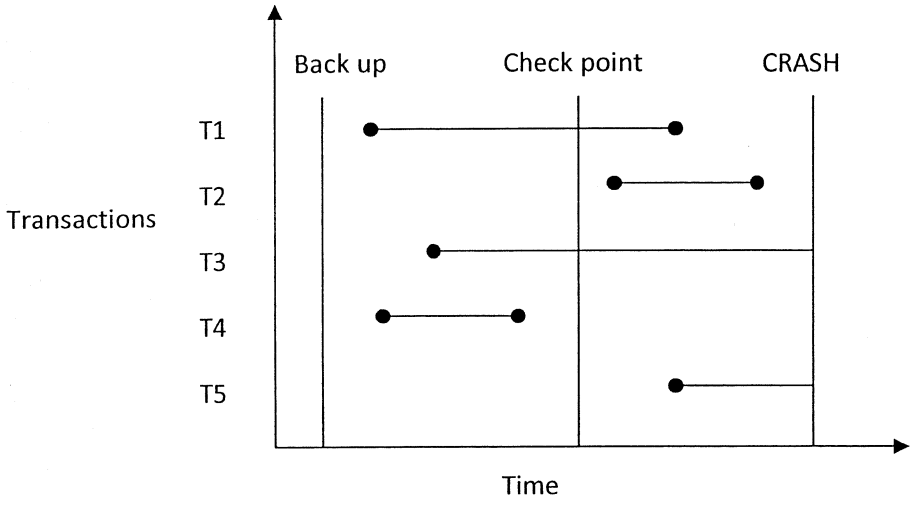
(4 marks)

ii. Follow deadlock detection approach to deal with deadlocks. Draw a transaction schedule and explain how three transactions are running. Show a waits-for graph if a deadlock cycle develops.

(4 marks)

Question 8 – Crash Recovery (20 marks)

Consider the diagram below:



(a.) What is the desired state that the database should be in after Crash Recovery? Briefly explain.
(5 marks)

(b.) Explain the *Write-Ahead Logging* (WAL) Protocol and its importance.
(4 marks)

(c.) How does WAL protocol assist to ensure *Atomicity* and *Durability* in a STEAL-NO FORCE approach?
(5 marks)

(d.) What are the roles of Analysis, Redo and Undo phases in ARIES?
(6 marks)