The Open University of Sri Lanka
Faculty of Engineering Technology
Department of Electrical & Computer Engineering

Study Programme	: Bachelor of Software Engineering Honors
Name of the Examination	: Final Examination
Course Code and Title	**: ECI5266/EEI5566 – Advanced Database Systems**
Academic Year	: 2019/2020
Date	: 25th July 2020
Time	: 1330-1630hrs
Duration	: **3 hours**

1. Read all instructions carefully before answering the questions.
2. This question paper consists of **Eight (8)** questions in **Ten (10)** pages.
3. Answer **Five (5)** questions **ONLY**.
4. Answer for each question should commence from a new page.
5. This is a Closed Book Test **(CBT).**
6. Answers should be in clear handwriting.
7. Do not use red color pen.

Question 1 – Database Programming (20marks)

Consider the database of a department store that includes the following tables:

ITEM (*ino*:integer, *iname*:varchar(10), *dept*:integer, *price*:real, *qty*:integer, *cost*:real)
EMP (*eno*:integer, *ename*:varchar(15), *salary*:integer, *comm*:real, *dept*:integer)
DEPT (*dno*:integer, *dname*:varchar(10), *manager*:integer)
SALES (*ino*:integer, *eno*:integer, *stime*:date, *qty*:integer, *price*:real)

The table ITEM contains the item number, name, department number, current price per unit, quantity in stock, and cost per unit for all items. The department number value in an item tuple indicates the department that is responsible for selling that item. The table EMP contains the employee number, name annual salary, sales commission, and the department number of every employee. The table DEPT contains the department number, name, and the employee number of the manager for each department. The SALES table records each sale made by an employee with item number, employee number, date and time of sale, quantity, and price. The primary keys of the relations are underlined.

Use the department store database schema and write answers to the following questions.

a) Write a T-SQL procedure named *insertsale* to insert a row into the SALES table given the input parameters of employee number, item number, and quantity. The sale date and time is the current date and time; the sale price is the current price of the item in the ITEM table.

(6 marks)

b) Write a T-SQL procedure named *updcomm* to update the commission of the employee, given the input parameters of employee number, item number and quality. The commission payable is 10% of the profit on the sale. The profit on a sale is the product of quantity and (price - cost) of the item.

(7 marks)

c) Write a trigger on the SALES table that will reduce the quantity of the relevant item in the ITEM table whenever a new row is inserted into the SALES table. It should also update the commission payable to the employee involved by using the procedure of question (b) above.

(7 marks)

Question 2 – Relational Algebra (20marks)

a) Consider the following relations R1 and R2:

R1

A	B	C
a1	b1	c1
a2	b2	c2
a3	b3	c3

R2

X	Y	Z
c2	y2	z2
c3	y3	z3
c4	y4	z4

Use the above relations to perform the following relational algebra operators and write output relations.

i. $R1 \bowtie_{R1.c = R2.x} R2$ (3 marks)

ii. $R1 ⟗_{R1.c = R2.x} R2$ (3 marks)

iii. $R1 ⟕_{R1.c = R2.x} R2$ (3 marks)

iv. $R1 ⟖_{R1.c = R2.x} R2$ (3 marks)

b) Consider the following relational schema.

Student (*<u>snum</u>, sname, major, level, age*)
Enrolled (*<u>snum, cname</u>*)
Class (*<u>cname</u>, time, room, lid*)
Lecturer (*<u>lid</u>, fname, deptid*)

Write the following queries in **relational algebra**.

i. Find the names of all Juniors (*level* = *JR*). (3 marks)

ii. Find the names (*cname*) of all classes that were either held in room H501 (*room* = *H501*) or have hundred or more students enrolled. (5 marks)

Question 3 – Object Relational Databases (20marks)

Consider the following object relational schema for a University database:

Object types:

course_t (cid: *integer*, title: *varchar(15),* credits_req: *integer*)

unit_t (uid: *integer*, credits: *integer*)

offering_t (unit: *ref unit_t*, semester: *number(1)*, year: *number(4)*)

enrolled_t (unitoffer: *ref offering_t*, mark: *integer*)

enrolled_list table of enrolled_t

student_t (sid: *integer*, name: *varchar(15),* phone: *varchar(10),* course: *ref course_t,* enrolments: *enrolled_list*)

Tables:

Courses of course_t (cid primary key)

Units of unit_t (uid primary key)

Offered of offering_t (unit not null references units)

Students of student_t (sid primary key, course references courses) Nested table enrolments store as enrolledlist_ntab

- The Courses table of course_t has attributes of course id (cid), title, and credit points required. It contains tuples for all courses. Units table of unit_t contains tuples for all units, and has the attributes unit number (uid), and number of credits.
- The Offered table of offering_t records the information of units offered in a particular semester and consists of attributes for unit reference, semester and year.
- The Students table of student_t contains student id, name, phone, course enrolled and the list of enrolled units (as an attribute of nested table type) for each student. Each tuple of the nested table contains a reference to unit offering, and the mark obtained by the student.
- 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 for the following query (use columns of REF type instead of joins to link tables):

For students who are enrolled in units offered during 2006, find the name of the student, course title and the average mark for the units taken in 2006.

(4 marks)

b) Add a new enrolment for student with *sid* 1256432 in the unit with *uid* 302636 in semester 2 of year 2006. Assume that the unit offering already exists in the Offered table and the given student also exists in the students table with other enrolments. The mark should be entered as null.

(6 marks)

c) It is required to add a member method called credcomp to calculate the number of credits completed by a student. Only units where student received a mark of at least 50 are counted as completed.

Write Oracle SQL statements to modify the object type student_t by adding this method specification.

(6 marks)

d) Using the method defined above, write an Oracle SQL statement to display for each course, the course title and the number of students who have completed the required credits for their courses.

(4 marks)

Question 4 – Disk, Files and Indexes (20marks)

a) What is the justification for using I/O costs as the main measure to compare different algorithms for evaluating relational operators?

(4 marks)

b) Briefly explain why it is often advantageous to do selections before joins in a query plan. How do early projections help during the query execution?

(4 marks)

c) Briefly describe what happens in external merge sort in first two passes.

(4 marks)

d) Briefly describe two main techniques used in RAID. What is the purpose of each technique? (Note : You do not have to explain the RAID levels)

(5 marks)

e) Explain the role of the Disk Space Manager in the database management system architecture?

(3 marks)

Question 5 – XML Databases (20marks)

Consider the following XML document that contains information on a collection of books.

```
<?xml version="1.0" encoding="UTF-8"?>
<books>
   <book category="JAVA">
      <title lang="en">Learn Java in 24 Hours</title>
      <author>Robert</author>
      <year>2005</year>
      <price>30.00</price>
   </book>
   <book category="DOTNET">
      <title lang="en">Learn .Net in 24 hours</title>
      <author>Peter</author>
      <year>2011</year>
      <price>40.50</price>
   </book>
   <book category="XML">
      <title lang="en">Learn XQuery in 24 hours</title>
      <author>Robert</author>
      <author>Peter</author>
      <year>2013</year>
      <price>50.00</price>
   </book>
   <book category="XML">
      <title lang="en">Learn XPath in 24 hours</title>
      <author>Jay Ban</author>
      <year>2010</year>
      <price>16.50</price>
   </book>
</books>
```

Note that the above XML document stored in Books(xText XML) table created in MS SQL Server and it contains an only single record.

a) Write an XQuery (FLWOR) expression to retrieve the *titles* of those books with a *price* greater than 30. (4 marks)

b) What would be the output of the following XQuery expression?

```
SELECT xText.query ('
               let $books := /books/book
               return <results>
                     {
                        for $x in $books
                        where $x/price>=30
                        order by ($x/price)[1]
                        return ($x/title, $x/price)
                     }
                     </results>
               ')
FROM Books
```

(4 marks)

c) Write an XQuery expression to produce the following output which is to retrieve the *titles* of books with a *price* greater than equal 30. And books are in increasing order of the prices.

```
<results>
  <title lang="en">Learn Java in 24 Hours</title>
  <price>30.00</price>
</results>
<results>
  <title lang="en">Learn XQuery in 24 hours</title>
  <price>50.00</price>
</results>
<results>
  <title lang="en">Learn .Net in 24 hours</title>
  <price>70.50</price>
</results>
```

(6 marks)

d) Write an XQuery expression to extract data from books XML document and create an HTML table containing the titles of all the books along with their respective prices. Use the following HTML tags in your answer.

```
<table>
  <tr>
    <th>Title</th>
    <th>Price</th>
  </tr>
  <tr>
    <td>Learn XPath in 24 hours</td>
    <td>16.50</td>
  </tr>
      … … …
  <tr>
    <td>Learn .Net in 24 hours</td>
    <td>70.50</td>
  </tr>
</table>
```

(6 marks)

Question 6 – Query Processing (20marks)

a) Why is query optimization important in a DBMS? Explain your answer.

(2 marks)

b) What are the stages in query optimization?

(2 marks)

c) Consider the following relation schemas for recording information on units, students and enrolments:

Students (sno, sname, age, course, phone)
Units (uno, uname, credit)
Enrolments (sno, uno, mark)

Students table consists of 1000 pages with 20 tuples per page, Units table has 50 pages of 50 tuples each, and Enrolments has 500 pages of 60 tuples each. There are 60 buffer pages.

Consider the query:

```
SELECT DISTINCT s.sno, s.sname
FROM students s, enrolments e
WHERE e.sno = s.sno
        and s.course= 'CS'
        and e.uno = 4533
        and e.mark≥50;
```

Assume that only 5% of Students tuples are in the CS course, 0.5% of Enrolment tuples are for the unit 4533 and 80% of those students passed (mark≥50) the unit. Students table has a clustered B+ tree index on <course,sno> and Enrolments has a clustered B+ tree index on <uno,sno>. Assume that course and sno columns take up 20% of each tuple in Students; uno and sno columns occupy 80% of space on each Enrolment tuple. Assume that the data entries of the leaf pages have the format *<search-key-field-value, record-id>*. To simplify your calculations, ignore the cost of storing record ids in the leaf pages.

i. Find the cost that performs each selection **without** using indexes.

(2 marks)

ii. Find the cost that performs selections using the clustered indexes.

(8 marks)

iii. Draw a query tree for the best-cost plan.

(6 marks)

Question 7 – Transactions and Concurrency Control (20marks)

a) Briefly explain the properties of a transaction.

(4 marks)

b) Briefly explain, what a *Serializable Schedule* is?

(1 mark)

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

(2 marks)

d) Consider a database with objects X and Y and assume that there are two transactions T1 and T2. Transaction T1 reads object X and Y and then writes object X. Transaction T2 reads objects X and Y and then writes objects X and Y. Both T1 and T2 commit after all read and write actions of them.

i. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a write-read conflict.

(2 marks)

ii. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a read-write conflict.

(2 marks)

iii. Give an example schedule with actions of transactions T1 and T2 on objects X and Y that results in a write-write conflict.

(2 marks)

iv. For each of the three schedules, show that Strict 2PL disallows the conflicts.

(2 marks)

e) Consider the following sequence of actions, listed in the order they are submitted to the DBMS. The Strict 2PL has been used for concurrency control.

T1:R(P), T2:W(Q), T2:W(P), T3:R(R), T3:R(Q), T1: W(R)

T_1	T_2	T_3
S(P)		
R(P)		
	X(Q)	
	W(Q)	
	X(P)	
	W(P)	
		S(R)
		R(R)
		S(Q)
		R(Q)
X(R)		
W(R)		

Assume that older transaction has higher priority always. The DBMS processes actions in the order shown.

i. Follow *wound-wait* policy to deal with deadlock in above schedule. Draw schedule again.

(3 marks)

ii. Follow deadlock detection approach to deal with deadlocks in above schedules. Draw a wait-for graph for the schedule given above to detect the deadlock and briefly explain how to identify the deadlock and remove.

(2 marks)

Question 8 – Crash Recovery (20marks)

a) Modern high performance database systems allow buffer frames to be stolen from uncommitted transactions. What problem does this create for the DBMS?

(3 marks)

b) In the context of a DBMS, explain what a database log is. In your answer, you should say what a log entry contains.

(4 marks)

c) What is the *Write-Ahead Logging* protocol?

(4 marks)

d) What is a *transaction table*, where is it stored, and what does it record?

(4 marks)

e) Briefly explain *UNDOing* and *REDOing* phases of crash recovery in DBMS.

(5 marks)