00048

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

Study Programme	: Bachelor of Software Engineering Honours
Name of the Examination	: Final Examination
Course Code and Title	: **EEI5466 Advanced Database Systems**
Academic Year	: 2021/22
Date	: 13th February 2023
Time	: 09.30-12.30hrs
Duration	: 3 hours

General Instructions

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

Question 1 – Database Programming **(20 marks)**

Consider a database scheme consisting of the following relations for recording information about students in a university.

Units (unitindex char(8), UnitName varchar(40), CreditPoints integer)
Students (ID char(8), FirstName varchar(15), LastName varchar(15) not null, BirthDate date)
Enrollments (Student char(8), Unit char(8), Year Number (4), Mark number (3))

The primary keys are underlined. A tuple in the units table consists of the unitIndex, unitName and the number of credit points for that unit.

A tuple in the Students table consists of the Student's ID, FirstName and LastName as well as their Birth Date. The enrollments table records information about enrollments of students in units. A tuple in the enrollments table consists of the Student (he enrolled), the Unit (the unit index of the unit), the year of the enrollment and the mark (if any) achieved for that unit. The unit attribute is a foreign key referring to the units table. Similarly, the student attribute is a foreign key referring to the student table.

a) Write appropriate SQL statement for the following.

   i. Find the units with less than 10 students enrolled in them. Report only the unit name. **(2 marks)**

   ii. For each student who has completed more than two units, print the student last name and number of Units obtained more than 75 marks. Marks record in the Enrollments table only after completing the unit. **(3 marks)**

b) It has been found that many users of the database frequently have a need to find out how many students are enrolled in a particular unit in a year, although different users are interested in different units.

   i. Produce a view called 'studentsByUnit' to make this query convenient. Include the unit name in addition to the other relevant attributes for this view. **(3 marks)**

   ii. Use this view to find out how many people are enrolled in "Database Systems" in 2013. **(2 marks)**

c) Time has passed since the view 'studentsByUnit' mentioned in part b) i. was set up, and it has been found that studentsByUnit is consuming significant computing resources because so many people are using it. It has been decided to reduce this problem by adding a new attribute called stuCount to the units table. Do this as follow.

   i. Add an attribute of type integer called stuCount to the Units table. **(1 mark)**

   ii. Write an update statement to initialize the values of the stuCount attribute so that for each unit the value of stuCount is the number of students enrolled in that unit. **(3 marks)**

   iii. Write a trigger to maintain the stuCount attribute in the event that an enrollment is **removed** from the enrollment table. **(6 marks)**

Question 2 – Object Relational Databases **(20 marks)**

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 queries (use columns of REF type instead of joins to link tables):

   i. Find the name and course title of students who have completed the total credits required for their course (i.e., the total credits of units in which the student obtained at least 50 marks is greater than or equal to the credits required). **(5 marks)**

   ii. For students who are enrolled in units offered during 2015, find the name of the student, course title and the average mark for the units taken in 2015. **(4 marks)**

b) Add a new enrolment for student with *sid* 1256432 in the unit with *uid* 302636 in semester 2 of year 2015. 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.
   (5 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)**

Question 3 – Relational Algebra **(20 marks)**

Consider the given schema for the following questions 11, 12, 13, 14 and 15 are given below:

Product (model, maker, type)
PC (model, speed, ram, hdsize, rd, price)
Laptop (model, speed, ram, hd, screen, price)
Printer (model, color, type, price)

The **Product** relation gives the manufacturer, model number and type (PC, laptop or printer) of various products.
The **PC** relation gives the model number, speed of the processor, the amount of RAM, the size of the hard disk, removable disk (CD or DVD), and the price.
The **Laptop** relation is similar, except that the screen size.
The **Printer** relation records the printer model, color printer (true, if so), the type (laser, ink-jet or bubble), and the price.

Which of the following queries in **relational algebra** is correct regarding the Relational Schema given above?

a) What PC models have a speed of at least 1000?
A
B
C
D
E None of the above **(4 marks)**

b) Which manufacturers make laptops with a hard disk of at least one giga-byte?
A
B
C
D
E *None of the above* **(4 marks)**

c) Find the model number and price of all products (of any type) made by manufacturer B.
A
B
C
D
E None of the above

(4 marks)

d) Find those manufacturers that sell Laptops, but not PC's.

A
B
C
D
E **(4 marks)**

e) Find those hard-disk sizes that use same hard disk size in two or more PC's.
A
B
C
D

E None of the above **(4 marks)**

Question 4 – Disk, Files and Indexing **(20 marks)**

a) You have a large file that is frequently accessed sequentially. Briefly explain how you would store the pages in the file on a disk for fast querying. **(2 marks)**

b) Consider a disk with a sector size of 512 bytes, a block size of 1024 bytes, 2000 tracks per surface, 50 sectors per track, 5 double-sided platters, average seek time of 10msec.

i. What is the capacity of a track in blocks? **(1 mark)**
ii. What is the capacity of a cylinder in blocks? **(1 mark)**
iii. What is the capacity of the disk in blocks? **(1 mark)**
iv. If disk platter rotates at 5400 rpm (revolutions per minute), what is the average rotational delay? **(2 marks)**
v. What is the access time to read two consecutive disk blocks? **(2 marks)**

c) An index on a database table can be either clustered or unclustered. From the following, choose the statements that are true.

A. A table can have only one unclustered index but may have any number of clustered indexes.
B. A table can have only one clustered index but may have any number of unclustered indexes.
C. Both B+-tree and hash indexes can be clustered.
D. Only B+-tree indexes can be clustered.
E. With a clustered index, the corresponding data pages are not always in sorted order.
F. None of the above. **(3 marks)**

d) Consider the following relation:

Emp (eid: integer, ename:string, age: integer, salary: float)

On this table, there is a dense clustered B+ tree index on *eid* and an unclustered B+ tree index on *age*.

i. The data records are stored in a heap file. In what order are the data records of Emp table stored and why? **(1 mark)**

ii. Consider the following query:
Select ename, salary
From emp
Where age>18;
If 95% of tuples satisfy the selection condition, what would be the best access path for processing this query? Justify your answer. **(3 marks)**

iii. Consider the following query:
Select age, avg(salary)
From emp
Group by age;
Would a clustered index on <age, salary> be useful? Explain briefly. **(4 marks)**

Question 5 – Query Processing **(20 marks)**

a) What are the steps in Query Processing? Explain each step. **(4 marks)**
b) Why does it cost more to use an unclustered index than a clustered index to retrieve tuples that satisfy a range selection? **(4 marks)**
c) Briefly describe what happens in external merge sort in first two passes. **(4 marks)**
d) Consider a relation named *parts* with primary key *pid* and 100,000 records stored 10 records per block or page. If 1% of records are randomly accessed by *pid* value every day, compare the costs in number of disk accesses between a hash index and a B+ tree index on *pid*. **(4 marks)**
e) Consider the following schema:

Item (i_num:integer; i_name:string, unit_price:float)

Order (o_num:integer, o_date:date, i_num:integer, qty:integer)

Consider the following query:

```
SELECT    Item.i_name
FROM      Item, Order
WHERE     Item.i_num = Order.i_num AND Order.o_num = '1234'
```

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. What query plan in (i.) would you suggest executing on a DBMS and briefly explain the query plan you consider? **(2 marks)**

Question 6 – Transactions and Concurrency Control **(20 marks)**

a) List the properties of a transaction. Briefly explain each of them. **(4 marks)**
b) Compare Serial Schedule vs Serializable Schedule? **(2 marks)**
c) Briefly explain the rules in Strict 2 Phase Locking (S2PL) Protocol. **(2 marks)**
d) Consider a database with objects A and B and assume that there are two transactions T1 and T2. Transaction T1 reads object A and B and then writes object A. Transaction T2 reads objects A and B and then writes objects A and B. Both T1 and T2 commit after all read and write actions of them.

i. Give example schedules with actions of transactions T1 and T2 on objects A and B that results in write-read conflict, read-write conflict and write-write conflict. **(6 marks)**
ii. For each of these 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 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.

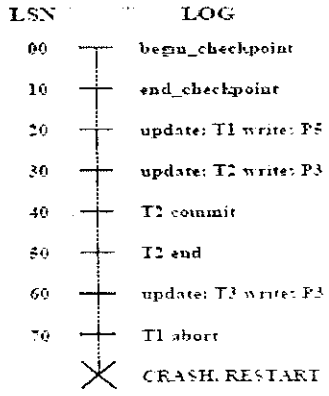
T1:R(X), T2:W(X), T2:W(Y), T3:W(Y), T1:W(Y),T3W(X), T1: Commit, T2: Commit, T3: Commit

Assume that older transaction has higher priority always.

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)**

Question 7 – Crash Recovery **(20 marks)**

a) Modern high performance database systems allow buffer frames to be stolen from uncommitted transactions. What problem does this create for the DBMS? **(5 marks)**
b) What is the Write-Ahead Logging protocol? **(5 marks)**
c) What is a transaction table, where is it stored, and what does it record? **(5 marks)**
d) Consider the execution shown in the figure.



Discuss how the Analysis and Redo phases of the ARIES algorithm relate to the schedule above. State if any assumptions you made. **(5 marks)**