UNIVERSITY OF COLOMBO, SRI LANKA

UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING

DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)

Academic Year 2021 – 3rd Year Examination – Semester 6

IT6405: Database Systems II

Structured Question Paper

18th March, 2022

(TWO HOURS)

To be completed by the candidate

BIT Examination Index No: ....

Important Instructions:

- The duration of the paper is **2 (Two) hours**.
- The medium of instruction and questions is English.
- This paper has **4 questions** and **18 pages**.
- **Answer all questions.** All questions carry **equal** marks.
- **Write your answers** in English using the space provided **in this question paper**.
- Do not tear off any part of this answer book.
- Under no circumstances may this book, used or unused, be removed from the Examination Hall by a candidate.
- Note that questions appear on both sides of the paper.
  If a page is not printed, please inform the supervisor immediately.
- Calculators are **not** allowed.
- *All Rights Reserved.*

Questions Answered

Indicate by a cross (×), (e.g. ☒) the numbers of the questions answered.

	Question numbers				
To be completed by the candidate by marking a cross (×).	1	2	3	4	
To be completed by the examiners:					

Index No ....

1) (a) Briefly describe two common uses of executing a Stored Procedure.

[2 marks]

ANSWER IN THIS BOX

- Execution of all complicated queries using stored procedures is faster since they are compiled and stored in the database server.
- Allowed to group all the required SQL statements into a procedure and execute them at once.
- Reusable and avoid repetition of code.
- Once compiled a stored procedure, it can be used in any number of applications. If any changes are required, stored procedures can be changed without touching the application code.
- Increase scalability by isolating application processing on the server.
- Its security is high since access to the data can be restricted by allowing users to manipulate the data only through stored procedures that execute with their definer's privileges.
- Reduce data transfer and communication cost between the client and server in certain situations.

Any two of the above answers and one mark per each correct answer.

(b) Briefly explain two differences between 'Functions' and 'Stored Procedures' with respect to database management systems.

[2 marks]

ANSWER IN THIS BOX

- The function must return a value but in Stored Procedure it is optional. Even a procedure can return zero or n values.
- Functions can have only input parameters whereas Procedures can have input or output parameters.
- Functions can be called from Stored Procedure whereas Procedures cannot be called from a Function.

Any two of the above answers and one mark per each correct answer.

(c) Consider that, a company maintains a table named ***product*** in the ***classicmodel*** database given below which keeps track of product details.

product (productCode, productName, productLine, productScale, productVendor, productDescription,, quantityInStock, buyPrice DECIMAL, MSRP DECIMAL)

The management of the company decided to update the current value of Manufacturer's Suggested Retail Price (*MSRP*) of a given product by adding a particular rate to the *buyPrice* of the product which is represented through the formular **MSRP = buyPrice + buyPrice * rate.** Write down the stored procedure ***update_MSRP*** with two input parameters, *productCode* and *rate* (value between 0% to 100%) to update the existing *MSRP* of the product relevant to the given *productCode*.

[13 marks]

ANSWER IN THIS BOX

```
CREATE PROCEDURE `classicmodels`.`update_MSRP` (IN proCode
varchar(15), IN rate Float) [ 3 marks]

BEGIN[ 1 mark]
declare updatedMSRP float; [ 1 mark]

Select buyPrice into updatedMSRP
from classicmodels.products
where productCode = proCode; [ 2 marks]

set updatedMSRP := updatedMSRP + (rate*updatedMSRP)/100;
                                   [ 3 marks]
UPDATE classicmodels.products
SET MSRP = updatedMSRP
WHERE productCode = proCode; [ 2 marks]

END [ 1 mark]
```

(d) Consider the table ***payments*** in the ***classicmodel*** database as given below which keeps track of customer payment details. The *amount* in the table represents the value to be paid by each customer. However, the payable amount (*payableAmount*) of each customer is calculated based upon the discount criteria adopted by the company.

payments (customerNumber, checkNumber, paymentDate, amount, payableAmount)

Write down a trigger named ***order_discount*** to set *payableAmount* based on the below discount criteria upon insertion of new record into the *payments* table.

Discount Criteria:

- 20% discount for the total payable amount if amount is >= 20,000 LKR

- 10% discount for the total payable amount if amount is >= 10,000 LKR and amount < 20,000 LKR

[8 marks]

ANSWER IN THIS BOX

```
CREATE TRIGGER order_discount     [2 marks]
before INSERT ON payments    [1 mark]
FOR EACH ROW
BEGIN    [1 mark for BEGING END]
      if (new.amount >= 20000)
      then
      SET new.amount := new.amount * 0.8;   [2 marks]
END IF;
      if (new.amount >= 10000 and new.amount < 20000 )
      then
      SET new.amount = new.amount * 0.9;   [2 marks]
END IF;
END;
```

2) (a) Briefly describe the following three (03) indexing types.

(i) Primary Index

(ii) Secondary Index

(iii) Clustering Index

[6 Marks]

ANSWER IN THIS BOX

(i) Primary index is defined on an ordered data file. The data file is ordered on a Key field.The key field is generally the primary key of the relation. **[2 Marks]**

(ii) Secondary index may be generated from a field which is a candidate key and has a unique value in every record, or a non-key with duplicate values. **[2 Marks]**

(iii) Clustering index is defined on an ordered data file. The data file is ordered on a non-key field. **[2 Marks]**

(b) Consider the following relational schema taken from ***university*** database.

DEPARTMENT (DeptId, DeptName, HeadofDept)

STAFF (EmpId, EmpName, Age, Designation, Salary, DeptId)

Note that each *HeadofDept* value in the *Department* relation represents an *EmpId*. The designation of more than 50% of staff members is "Lecturer".

Given below are three (03) SQL queries executed using three (03) different query plans.

- Suggest the most suitable index file organization for each query. Consider using *index only plans* where necessary.
- Justify your suggestion for each query.

(i)
```
SELECT AVG(Salary)
FROM STAFF
WHERE Age >35;
```

(ii)
```
SELECT S.EmpId, D.DeptName
FROM STAFF S,DEPARTMENT D
WHERE D.DeptId=S.DeptId AND S.Designation= "Lecturer";
```

(iii)
```
SELECT COUNT(*)
FROM STAFF
WHERE DeptId= "ISE";
```

[6 Marks]

ANSWER IN THIS BOX

(i) Create a dense **clustered B+ tree** index **on <DeptId,Salary>** of the STAFF table. Creating index on non-key attributes will let us use **index only plan**. Clustered tree is better since where clause requires a **range query.** **[2 Marks]**

(ii) **Hash index on D.DeptId** and clustered **B+ tree index** on <S.EmpId,S.Designation>. Since majority are lecturers, the index should be clustered. Can use index only plan to retrieve data. **[2 Marks]**

(iii) Hash index on DeptId. Can use index only plan to retrieve data. **[2 Marks]**

(c) Assume that you have a file with 600,000 records (r), which is ordered by its key field and each record of this file is fixed length and unspanned. Record length (R) is 100 bytes and block size (B) is 4096.

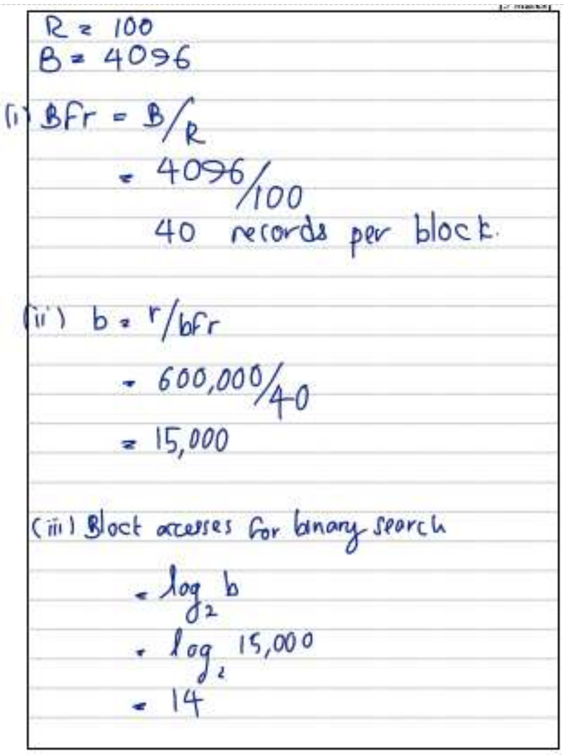
(i) What is the blocking factor?

(ii) How many blocks are required to store this file?

(iii) Calculate the number of block accesses required when performing a binary search on this file and access data.

[5 Marks]

ANSWER IN THIS BOX



$R = 100$

$B = 4096$

(i) $Bfr = B/R$

$= 4096/100$

40 records per block.

(ii) $b = r/bfr$

$= 600{,}000/40$

$= 15{,}000$

(iii) Block accesses for binary search

$= \log_2 b$

$= \log_2 15{,}000$

$= 14$

(d) Suppose a ***library*** database has the following schema.

Student (StuId, Name, Major)

Borrowings (StuId, Isbn, BorrowDate, DueDate)

Books (Isbn, Author, Title)

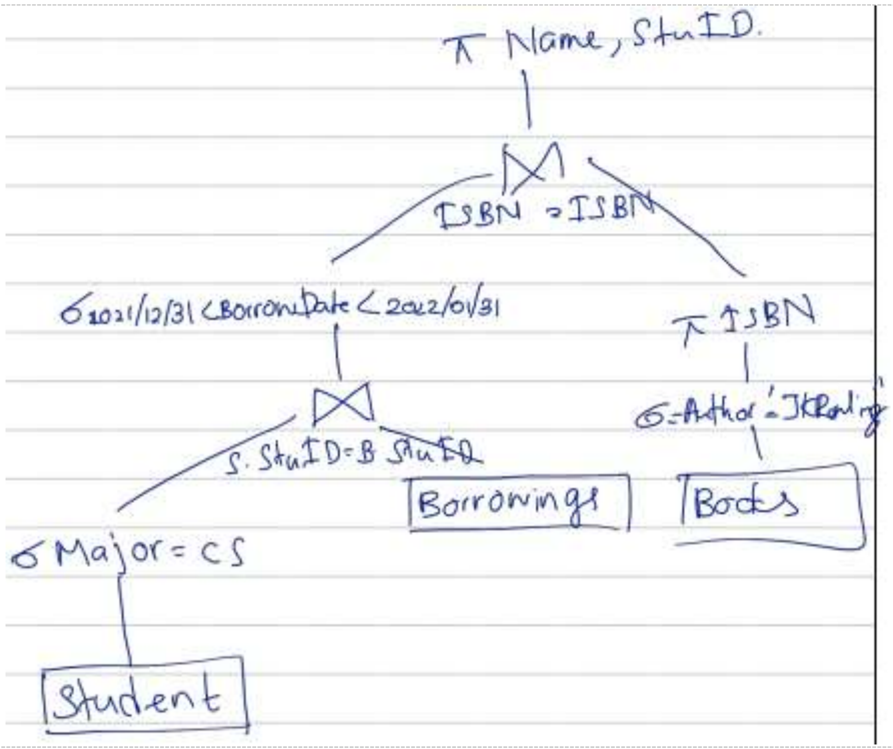
Consider the following query:

π Name, StuId (σ Major= “ComputerScience” AND 2021/12/01<BorrowDate <2022/1/31 AND Author = “JKRowling” (Student ⋈ Borrowings ⋈ Book))

(i) Draw the optimized query tree for the query given above.

[5 Marks]

ANSWER IN THIS BOX



(ii) Write the optimized query in relational algebraic expression for the above query tree in d (i).

[3 Marks]

ANSWER IN THIS BOX

$\pi_{Name, StuId}((\sigma_{2021/12/31 < BorrowDate < 2022/01/31}(\sigma_{major = 'cs'} Student \bowtie Borrowings)) \bowtie \sigma_{Author = 'JKRowling'} Books)$

3) (a) State the properties of a transaction and briefly explain each property.

[4 Marks]

ANSWER IN THIS BOX

[1 Mark for each]

1.Atomicity: A transaction is an atomic unit of processing. It is either performed in its entirety or not performed at all.

2. Consistency preservation: A correct execution of the transaction must take the database from one consistent state to another.

3. Isolation: A transaction should not make its updates visible to other transactions until it is committed.

4. Durability (Permanency): Once a transaction changes the database and the changes are committed, these changes must never be lost because of subsequent failures.

(b) State four (04) problems which may occur in concurrent Transaction Processing.

[4 Marks]

ANSWER IN THIS BOX

[1 Mark for each]

1.Lost Update Problem

2.Temporary Update/ Dirty Read

3.The Incorrect Summary Problem

4.The Unrepeatable Read Problem

(c) Identify the problems that would result in the transactions given below (i)-(ii).

[4 Marks]

(i) X= 80; Y=100; N=5; M=4

T1	T2
READ (X) X=X+N	

	READ(X)
	X=X+M

WRITE(X)
READ(Y)	

	WRITE (X)

Y=Y+N
WRITE(Y)	

(ii) X= 80; Y=100; N=5; M=4

T1	T2

READ (X)
X=X-N
WRITE(X)	

	READ(X)
	X=X+M

WRITE(X)
READ(Y)	

	WRITE (X)
READ(Y) ROLLBACK	

ANSWER IN THIS BOX

[2 Marks for each]

i-Lost Update

ii- Dirty Read

(d) (i) Consider the given schedule S for transactions T1, T2 and T3. Note that each ri and wi denote read, write operations belong to transaction Ti.

S: r1(X); r2(Y); r3(Z); w2(Y); w1(X); w3(X); r2(X); w2(X)

What is the equivalent serial schedule for the above schedule S?

[2 Marks]

ANSWER IN THIS BOX

T1 -> T3 -> T2

(ii) Consider a schedule S with three (03) transactions T1, T2 and T3 as given below. Note that each ri, wi and ci denote read, write and commit operations that belong to transaction Ti.

S: r1(X); w1(X); r1(Y); r1(Z); c1; r2(X); w2(X); r2(Z); w2(Z); c2; r3(Y); w3(Y); r3(Z); w3(Z); c3;

Is the schedule S a serial schedule? Explain the answer.

[3 Marks]

ANSWER IN THIS BOX

Answer: Yes **[1 mark]**

One transaction starts only after the commit of the previous transaction. In each transaction, it completes all the operations consecutively and start the next transaction. Therefore, S is a serial schedule. T1->T2->T3 is the serial schedule. **[2 marks]**

(e)

(i) Consider a schedule S with two transactions T1 and T2 as given below;

S: r1(X); r2(X); w1(Y); w2(Y); c1; c2;

Note that each ri, wi and ci denote read, write and commit operations that belong to transaction Ti.

Are there conflicting operations in this schedule? Explain your answer.

[3 Marks]

ANSWER IN THIS BOX

In this schedule, last 2 operations are conflicting as those two

- access same data item y
- operations are from 2 transactions.
- both of the operations are write operations

[1 mark for each]

(ii) Check whether the given schedule over transactions T1, T2, T3 is serializable by drawing a precedence graph. Justify your answer.

S=r3(Y);r3(Z);r1(X);w1(X);w3(Y);w3(Z);r2(Z); r1(Y);w1(Y);r2(Y);w2(Y);r2(X);w2(X);

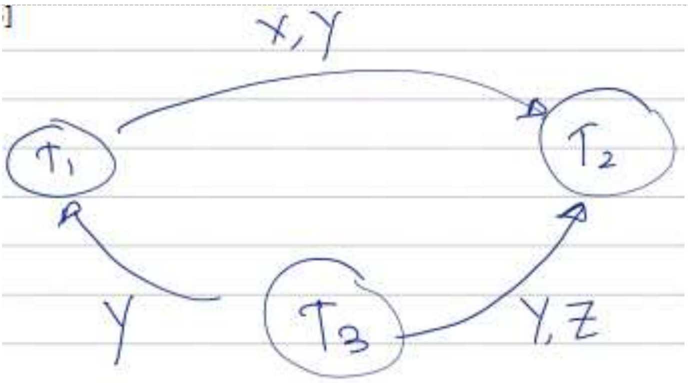
[5 Marks]

ANSWER IN THIS BOX

No cycles in the graph. **[1 mark]**

Therefore, schedule is serializable **[1 mark]**

Graph **[3 marks]**



4) (a) List down two (02) factors that influence the decision to use data replication in databases.

[2 marks]

ANSWER IN THIS BOX

(1 mark for any correct factor → max = 2 marks)

Database size

Usage frequency

High costs, including those for performance, software overhead and management.

Make data available in spite of the failure of one site.

Faster query execution

Increase parallelism

(b) Given below is a set of relations taken from a database of a private hospital. This hospital has four (04) main departments named HR, Finance, Admissions and Clinical. Each department is given the rights to maintain data about their employees through replicating data in their divisions. Under the Admission department, there are two sub divisions for Inpatients and Outpatients. Assume that all the departments and divisions have their own database and maintain their data as a synchronized distributed database system.

Employee *(Emp_ID, E_Name, Address, DOB, Designation, Joined_Date, Dept_Name, Salary)*
Patient *(P_ID, P_Name, DOB, Address, Contact_No, Medical_History, Admission_Status, Admission_Date)*

The following rules are applicable when sharing data with departments.

- Each department has the right to maintain data about their employees through replicating data in their departments except the salary information of each employee. However, the salary information of all the employees is handled by the HR department only.

- The admission department has access to patient information except for the medical history. The inpatient division in the admission department handles the patient details where admission status = 'Inpatient' and outpatient division handles the patient details where admission status = 'Outpatient' on *Patient* relation. Medical history of each patient is only visible to the Clinical department.

Assume that based on the above rules, a distributed database has to be designed. List the fragmented relations for each department and express the fragmentation conditions using relational algebra for each fragment.

[9 marks]

ANSWER IN THIS BOX

[1 mark each for the each correct fragmentation written using relational algebra → max = 9 marks]

HR department

Emp1 = π Emp_ID, Salary (Employee)

Emp2 = π Emp_ID, E_Name, Address, DOB, Designation, Joined_Date, Dept_Name (Employee)

Finance department

Emp3 = σ Dept_Name = "Finance" (Emp2)

Admissions department

Emp4 = σ Dept_Name = ”Admission” (Emp2)

Patient1 = π P_ID, P_Name, DOB, Address, Contact_No, Admission_Status, Admission_Date (Patient)

Clinical department

Emp5 = σ Dept_Name = ”Cilinical” (Emp2)

Patient2 = π P_ID, Medical_History (Patient)

Inpatient division

Patient3= σ Admission_Status = ‘Inpatient’ (Patient1)

Outpatient division

Patient4= σ Admission_Status = ‘Outpatient’ (Patient1)

(c) Identify the types of fragmentation you applied for each fragment in the section 4) (a) and indicate how you would reconstruct the original relation from those fragmentations.

[8 marks]

ANSWER IN THIS BOX

Fragmentations

(1 mark for identifying each correct fragmentation method → 2 marks)

Employee is vertically fragmented to Emp1 and Emp2.

Emp2 is horizontally fragmented to Emp3, Emp4, Emp5.

(1 mark for identifying each correct fragmentation method → 2 marks)

Patient is vertically fragmented to Patient1 and Patient2.

Patient1 is horizontally fragmented to Patient3 and Patient4 based on Admission_Status.

Reconstructions

(1 mark for identifying each correct reconstruction method → 2 marks)

Emp3 Ư Emp4 Ư Emp5 → Emp2

Emp1 ∞ Emp2→ Employee

(1 mark for identifying each correct fragmentation method → 2 marks)

Patient3 Ư Patient4 → Patient1

Patient1 ∞ Patient2 → Patient

(d) Consider the relational schema of ***Employee*** relation. Let E1, E2, E3 be fragmentations of Employee defined as follows.

Employee (Emp_ID, E_Name, Address, DOB, Designation, Joined_Date, Dept_Name, Salary)

E1 = π Emp_ID, E_Name, Address, DOB, Joined_Date, Dept_Name (σ Salary≥100,000 (Employee))

E2 = π Emp_ID, E_Name, Address, DOB, Joined_Date, Dept_Name (σ Salary≤100,000 (Employee))

E3 = π Designation, Dept_Name (Employee)

Identify whether the given fragmentations have violated the correctness rules of fragmentation? Justify your answer against each correctness rule.

[6 marks]

ANSWER IN THIS BOX

(2 mark for justifying each correctness rule correctly → 6 marks)

Reconstruction is violated because E3 does not contain the primary key, thus these three fragments (E1, E2, E3) could not be reconstructed into Employee, because there would be a lack of mapping.

Completeness would be fine, as all tuples and columns are stored.

Disjointness also violated – in E1 and E2, tuples where Salary=100,000 appear to be duplicated.

*****

Index No ………………………….