TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

ACADEMIC YEAR 2023/2024

MAY/JUNE EXAMINATION

BACS3183 ADVANCED DATABASE MANAGEMENT

FRIDAY, 7 JUNE 2024 TIME: 9.00 AM – 11.00 AM (2 HOURS)

BACHELOR OF COMPUTER SCIENCE (HONOURS) IN DATA SCIENCE

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN INTERNET TECHNOLOGY
BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN SOFTWARE SYSTEMS DEVELOPMENT

BACHELOR OF SOFTWARE ENGINEERING (HONOURS)

Instructions to Candidates:

Answer **ALL** questions. All questions carry equal marks.

Question 1

a) There are four types of information under the *source* category. List and provide an example of each type within TAR UMT environment. (8 marks)

b) Explain the **FIVE (5)** types of integrity constraint implemented on Oracle DBMS. (10 marks)

c) With reference to the TAR UMT' II Sisters Restaurant environment, illustrate and explain (using ERD, Crow's Foot notation) how a "many-to-many" relationship is modelled (exclude all attributes). Many-to-many relationship need to be resolved and you must indicate the strong or weak relationship lines. (3 + 4 marks)

[Total: 25 marks]

Question 2

The *Beautiful Hall Renting Company's* database is shown as follows:

Staff (StaffIC, StaffName, StaffAddress, StaffContact, StaffGender, StaffPosition, StaffSalary)
Member (MemberID, MemberName, MemberAddress, MemberContact, MemberGender)
HallBooking (MemberID*, HallID*, Purpose, BookingDate, BookingTime, StaffIC*)
Hall (HallID, HallName, HallSize*)
HallRentingCharge (HallSize, FeePerHour)

Note: member or staff gender is either 'M' for male or 'F' for female.

a) Write a ***relational algebra*** statement for each of the following questions:

(i) List out all male members (ID, name and address) who are staying at '*Petaling Jaya*'. (3 marks)

(ii) List out all admin staff (IC, name and gender) whose salary is more than or equal to RM2000. (3 marks)

(iii) List out all members (ID, name and contact) who had made a booking on December 2023 for '*Galaxy Hall*'. (6 marks)

(iv) List out the total number of member bookings for each hall (ID and name). (4 marks)

Question 1 a)

Primary information
- It is the original source document.
- For example, experimental data from university science lab.

Secondary information
- It is the processed primary sources or second-hand version.
- For example, student's essay discussing previous findings

Internal information
- It is gathered from various source within internal companies such as different departments.
- For example, examination schedule from the Academic Affairs department.

External information
- It is gathered outside the company such as interviewing customers or analyzing published data.
- For example, news article from a website about TAR UMT's new buildings.

Question 1 b)

NOT NULL
- Prohibits database value from being null.

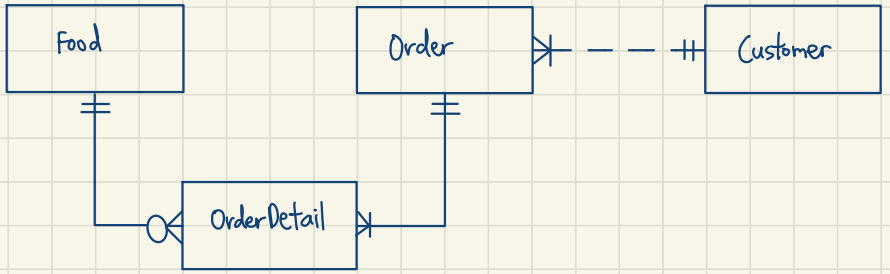
UNIQUE
- Prohibits multiple rows from having the same value in the same column.

PRIMARY KEY
- Combines NOT NULL and UNIQUE constraints in one single declaration.

FOREIGN KEY
- Requires the value in one table to match the value in another table.

CHECK
- Requires the column value in database to satisfy specified condition.
- For example, CHECK (status IN (`Active', `Inactive'))

Question 1 c)



- Each food may be listed in zero-to-many orders and each order may consist of one-to-many foods.

- Each order can be made by one and only one customer and each customer can make one-to-many orders.

Question 2 a)

(i) π MemberID, MemberName, MemberAddress (σ MemberGender = `M' $\wedge$ MemberAddress LIKE `% Petaling Jaya%' (Member))

(ii) π StaffIC, StaffName, StaffGender (σ StaffSalary $\geq$ 2000 $\wedge$ StaffPosition = `Admin' (Staff))

(iii) π MemberID, MemberName, MemberContact (Member) $\bowtie$ (Member.MemberID = HallBooking.MemberID (σ BookingDate $\geq$ `01/12/2023' $\wedge$ BookingDate $\leq$ `31/12/2023' (HallBooking)) $\bowtie$ (HallBooking.HallID = Hall.HallID (σ HallName = `Galaxy Hall' (Hall))))

(iv) π HallID, HallName (Hall) $\bowtie$ Hall.HallID = HallBooking.HallID $\Im$ COUNT MemberID (HallBooking)

Question 2 (Continued)

b) Refer to following SQL statement:

```
SELECT H.HallID, HallName, BookingDate, MemberID
FROM Hall H, HallBooking HB
WHERE H.HallID = HB.HallID AND BookingDate BETWEEN '01/01/2023' AND
      '31/12/2023'
ORDER BY HallName;
```

On what attributes should indexes be defined to speed up this query? Give reasons for each attribute selected. (9 marks)

[Total: 25 marks]

Question 3

Given the **StudentResult** table as follows:

StudentID	StudentName	ProgID	ProgName	CourseID	CourseName	LevelID	LevelDesc	ExamDate	Mark	Status
S1111	Alice Ng	P0001	RIT	C3001	Database	L002	Standard	11/03/2023	25	Fail
S1111	Alice Ng	P0001	RIT	C3001	Database	L002	Standard	08/11/2023	43	Fail
S1111	Alice Ng	P0001	RIT	C3001	Database	L002	Standard	10/06/2024		
S1111	Alice Ng	P0001	RIT	C3002	Big Data	L003	Medium	15/03/2023	52	Pass
S2222	Thomas Lee	P0001	RIT	C3001	Database	L002	Standard	11/03/2023	35	Fail
S2222	Thomas Lee	P0001	RIT	C3001	Database	L002	Standard	08/11/2023	50	Pass
S2222	Thomas Lee	P0001	RIT	C3002	Big Data	L003	Medium	15/03/2023	67	Pass
S3333	Penny Lim	P0002	RDS	C3002	Big Data	L003	Medium	15/03/2023	30	Fail
S3333	Penny Lim	P0002	RDS	C3002	Big Data	L003	Medium	19/11/2023	55	Pass
S4444	Julie Ong	P0002	RDS	C3003	NoSQL	L004	Advanced	23/11/2023	70	Pass

Table 1: Details of StudentResult Table

a) Normalise Table 1 to a set of Third Normal Form (3NF) relations. Your answer should show all the three stages of normalisation (1NF, 2NF and 3NF) by using the Database Design Language format (underline all primary keys, composite keys and use an * to indicate the foreign keys). State the functional dependency/dependencies that is/are removed from second and third Normal Form. Besides that, 1NF must be divided into repeating and non-repeating group relations from its original 1NF table. (16 marks)

b) Based on the sample data shown in the **StudentResult** table above, provide a specific example for insertion, modification and deletion anomalies. (9 marks)

[Total: 25 marks]

Question 2 b)

- HallID in Hall table because it is a primary key and the index will enforce the uniqueness of the key. HallID in Hall and in HallBooking table are used in a WHERE clause for joining the Hall and HallBooking tables, so it is logical to create an index on HallID in HallBooking as well. It can help for quickly match the hall in HallBooking for each HallID.

- HallName in Hall table because it is a nonkey cluster attribute used to sort the records faster.

- BookingDate in HallBooking table because it is a nonkey cluster attribute used to filter the records within the specified range of date in shorter time.

- MemberID in HallBooking table because it is a nonkey cluster attribute used to quickly find out the specified students' records without scanning the whole table.

- HallID, BookingDate, MemberID in HallBooking table because it is concatenated primary key and the index would enforce uniqueness of the key.

Question 3 a)

1NF

StudentResult (StudentID , StudentName, ProgID , ProgName, CourseID , CourseName, LevelID, LevelDesc, ExamDate , Mark , Status)

Student (StudentID, StudentName, ProgID, ProgName)
StudentResult (StudentID*, CourseID , CourseName, LevelID, LevelDesc, ExamDate , Mark , Status)

2NF

CourseID → CourseName , LevelID, LevelDesc (Partial dependency)

Student (StudentID , StudentName, ProgID, ProgName)
Course (CourseID , CourseName, LevelID, LevelDesc)
StudentResult (StudentID* , CourseID* , ExamDate , Mark , Status)

3NF

ProgID → ProgName (Transitive dependency)
LevelID → LevelDesc (Transitive dependency)

Student (StudentID , StudentName, ProgID*)

Programme (ProgID , ProgName)
Course (CourseID , CourseName , LevelID*)
Level (LevelID , LevelDesc)
StudentResult (StudentID* , CourseID* , ExamDate , Mark , Status)

Question 3 b)

Insertion Anomaly :
It is not possible to insert a new student record in the StudentResult table unless the student has participated in examination.

Modification Anomaly :
When we update the StudentName of record `Alice Ng' (S1111) from `Alice Ng' to `Alice Tan', we also have to update the StudentName value in other similar rows related to this student `Alice Ng' (S1111) to ensure data consistency.

Deletion Anomaly :
When we delete the record 'Julie Ong' (S4444) from StudentResult table, the course detail of record `NoSQL' (C3003) will also be deleted.

Question 4

a) Based on the following set of CakeID for the **Cake** table as shown in *Table 2*:

CakeID	CakeName	CakePricePerKg
201	Tiramisu Cake	45.00
202	Red Velvet Cake	35.00
203	Black Forest Cake	38.00
204	Marble Cake	25.00
205	Vanilla Cake	30.00
206	Chocolate Cake	30.00

Table 2: Cake Table

(i) Construct a *B+-tree* final structure with degree of 3 (6 marks)

(ii) Construct a *B+-tree* final structure with degree of 4 (4 marks)

(iii) Construct a *B+-tree* final structure with degree of 5 (3 marks)

b) The Sunlight Bubble Tea Shop has been serving bubble teas locally for over 10 years. The company currently has 5 outlets in different location. The company has total number of 20 staffs (4 staffs allocate for an outlet, each staff taking turn to serve at the 5 outlets). Determine the distribution plan for each relation' records on the following bubble tea shop database:

Staff (StaffID, StaffName, StaffAddress, StaffContact, StaffDOB, StaffPosition, StaffSalary)
Outlet (OutletID, OutletAddress, OutletContact)
BubbleTea (BubbleTeaID, BubbleTeaName, SugerUsed, BubbleUsed, IngredientUsed)
Sales (SalesNo, SalesDate, BubbleTeaID*, OutletID*, StaffID*, Quantity)

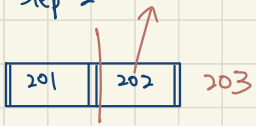
(12 marks)

[Total: 25 marks]

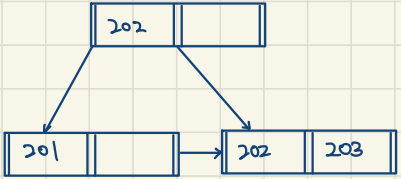
Question 4 a) (i)

Step 1

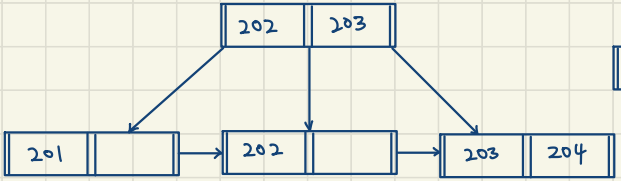
Step 2



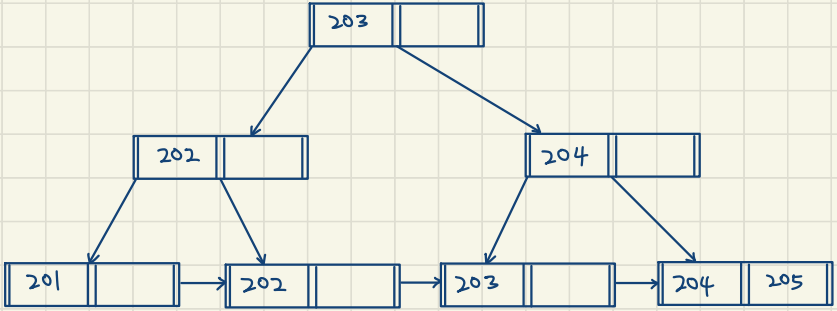
Step 3



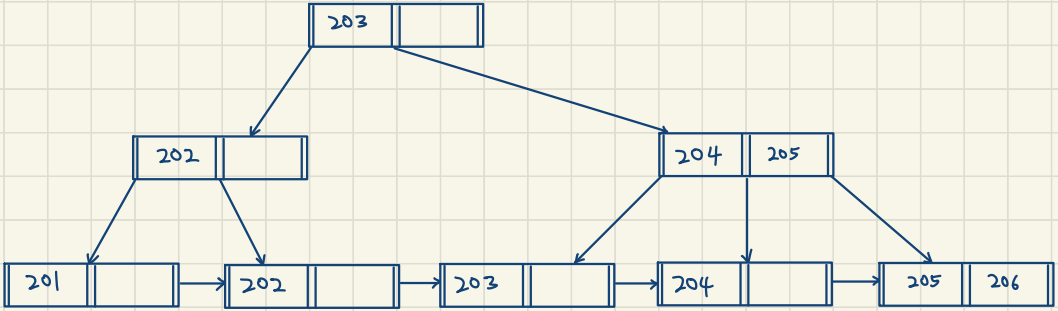
Step 4



Step 5



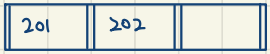
Step 6



Question 4 a) (ii)

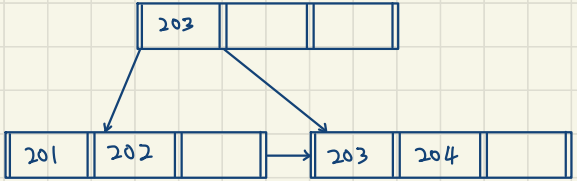
Step 1

Step 2

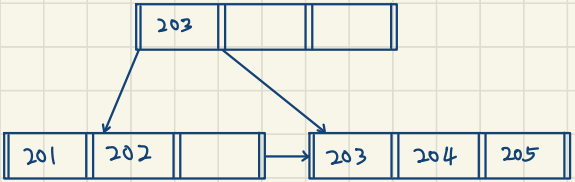


Step 3

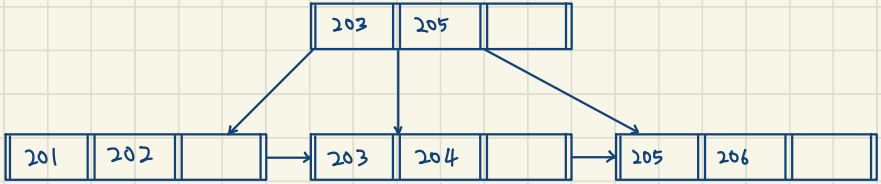
Step 4



Step 5



Step 6



Question 4 a) (iii)

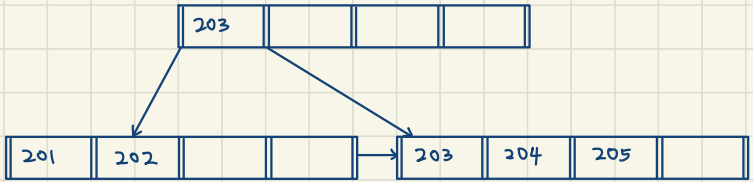
Step 1

Step 2

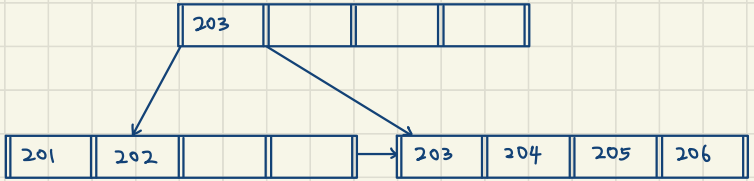
Step 3

Step 4

Step 5



Step 5



Question 4 b)

- Staff records are replicated since they can be served at any outlets and are rarely updated.
- Outlet records are replicated since they can be used at any outlets and are rarely updated.
- BubbleTea records are replicated since they are updated only once a new bubble tea flavour launch and must be easily accessible from all locations.
- Sales records are stored (partitioned) at the outlet where each outlet manages their own sales.