

TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

ACADEMIC YEAR 2023/2024

OCTOBER EXAMINATION

COMPUTER SCIENCE BACS3183
ADVANCED DATABASE MANAGEMENT

MONDAY, 23 OCTOBER 2023

TIME: 2.00 PM – 4.00 PM (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.

BACS3183 ADVANCED DATABASE MANAGEMENT**Question 1**

- a) Information can be categorised by *source*, *nature*, *level* and *time*. State and provide an example of each categorisation information by **FOUR (4)** *sources* within TAR UMT environment. (8 marks)
- b) Explain what *Transaction Processing Monitor (TPM)* is and its **FIVE (5)** *functions* that maintain the consistency within online processing environment. (12 marks)
- c) With the aid of a diagram and an example within TAR UMT environment, describe how a many-to-many relationship is modelled in an ERD (Entity Relationship Diagram) using crow's foot notation. (5 marks)

[Total: 25 marks]

Question 2

Given the Health Clinic Malaysia database as follows:

Patient (PatientID, PatientName, PatientGender, PatientContact, PatientAddress)
DiseaseExamination (PatientID*, ExamDate, DiseaseID*, HospitalID*, ExamCost)
Hospital (HospitalID, HospitalName, HospitalAddress, HospitalContact, HospitalEmail)
Disease (DiseaseID, DiseaseName, DiseaseDesc, DiseaseCategory, TreatmentDetails)

Note: Patient gender either 'M' for male or 'F' for female.

- a) Write a *relational algebra* statement for each of the following questions:
- List out all male patient (ID, name, address and contact). (2 marks)
 - List out all female patient (ID, name, address and contact) who is staying at 'Selayang Jaya'. (3 marks)
 - List out all patient (ID, name, gender and contact) who got the 'Dengue Fever' (disease name). (6 marks)
 - List out the total number of diseases for each disease category. (2 marks)
 - List out the total amount of exam cost for each type of disease (DiseaseID and DiseaseName). (3 marks)

Question 1

a) Source

- Example of internal information is students' class schedules.

Nature

- Example of quantitative information is TAR UMT campus has 3 canteens.

Level

- Example of operational information is each class with different courses will be assigned to different lecturers to teach students of that class.

Time

- Example of historic information is the past year exam papers of different courses.

b) - Transaction routing

- ↳ TPM increases scalability by directing transactions to specified DBMS.

- Manage distributed transactions

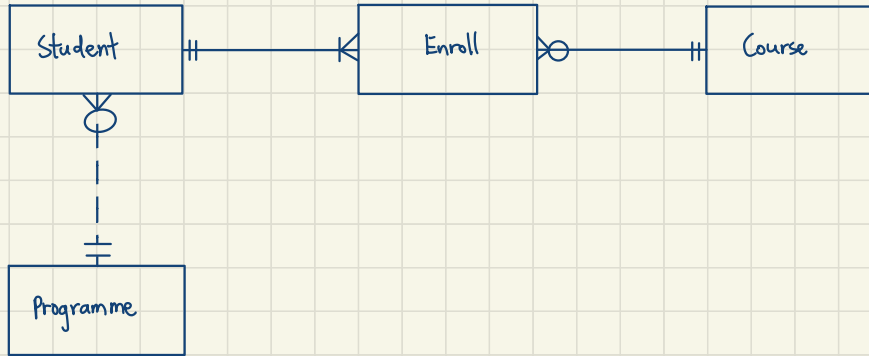
- ↳ TPM manages transactions that requires access to data that held by different DBMSs.

- Load balancing

- ↳ TPM balances user requests across multiple DBMSs on one or more computers.

- Increased reliability
 - ↳ TP monitor acts as transaction manager.
 - ↳ It performs necessary operations for maintaining database consistency.
 - ↳ When a database fails, TP monitor will resubmit the transaction to other DBMSs or hold the application until DBMS becomes available again.
- Funneling
 - ↳ TP monitor establishes connections to DBMS when required
 - ↳ It funnel user requests through these connections.

c)



- Each programme can be assigned to zero-to-many students and each student can apply for one and only one programme.
- Each student may enroll one-to-many courses and each course can be enrolled by zero-to-many students

Question 2 a)

(i) π PatientID, PatientName, PatientAddress, PatientContact (σ PatientGender = 'M' (Patient))

(ii) π PatientID, PatientName, PatientAddress, PatientContact (σ PatientGender = 'F' \wedge PatientAddress LIKE '%Selayang Jaya%' (Patient))

(iii) π PatientID, PatientName, PatientGender, PatientContact (Patient) \bowtie (Patient.PatientID = DiseaseExamination.PatientID (DiseaseExamination)) \bowtie (DiseaseExamination.DiseaseID = Disease.DiseaseID (σ DiseaseName = 'Dengue Fever' (Disease)))

(iv) π DiseaseCategory γ COUNT DiseaseID (Disease)

(v) π DiseaseID, DiseaseName (Disease) \bowtie Disease.DiseaseID = DiseaseExamination.DiseaseID γ SUM ExamCost (DiseaseExamination)

BACS3183 ADVANCED DATABASE MANAGEMENT**Question 2 (Continued)**

- b) Write the *SQL commands* to fulfill the requirements specified for the following scenarios:
- (i) All users of the database can view the patient information, but only Alice (registration supervisor) can update the patient address and contact. (4 marks)
 - (ii) Cecelia (registration manager) can do whatever she wishes to the patient information. At the same time, Cecelia should be able to grant the privilege to others. (3 marks)
 - (iii) Recently, Alice has resigned from the Health Clinic Malaysia, and she is no longer permitted to do anything on patient information. (2 marks)

[Total: 25 marks]

Question 3Given the **ProjectAssign** table as follows:

ProjectID	ProjectName	CategoryID	CategoryName	AssignDate	StaffID	StaffName	JobID	JobName	Hours	Remarks
P3111	Skylark	C001	Web Application	08/01/2023	S1020	Alice	J008	Team Lead	100	Phase I
P3111	Skylark	C001	Web Application	08/01/2023	S1323	Eric	J002	Programmer	250	Phase I
P3111	Skylark	C001	Web Application	11/04/2023	S1020	Alice	J008	Team Lead	100	Phase II
P3111	Skylark	C001	Web Application	11/04/2023	S1323	Eric	J002	Programmer	150	Phase II
P3322	Restaurant A	C002	Shop System	15/05/2023	S2323	Julia	J002	Programmer	80	Initial Stage
P3322	Restaurant A	C002	Shop System	12/07/2023	S2323	Julia	J002	Programmer	100	Final Stage
P3888	Slot Game	C003	Mobile App	12/09/2023	S3388	Kelvin	J003	App Developer	120	Need license

Table 1: Details of ProjectAssign 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 **ProjectAssign** table above, provide a specific example for insertion, modification and deletion anomalies. (9 marks)

[Total: 25 marks]

Question 2 b)

(i) GRANT SELECT (PatientID, PatientName, PatientGender, PatientContact, PatientAddress) ON Patient TO PUBLIC ;
GRANT UPDATE (PatientAddress, PatientContact) ON Patient TO Alice ;

(ii) GRANT ALL PRIVILEGES ON Patient TO Cecelia WITH GRANT OPTION;

(iii) REVOKE ALL PRIVILEGES ON Patient FROM Alice;

Question 3 a)

1NF

ProjectAssign (ProjectID, ProjectName, CategoryID, CategoryName, AssignDate, StaffID, StaffName, JobID, JobName, Hours, Remarks)



Project (ProjectID, ProjectName, CategoryID, CategoryName)

ProjectAssign (ProjectID*, AssignDate, StaffID, StaffName, JobID, JobName, Hours, Remarks)

2NF

StaffID → StaffName, JobID, JobName (Partial dependency)

Project (ProjectID, ProjectName, CategoryID, CategoryName)

Staff (StaffID, StaffName, JobID, JobName)

ProjectAssign (ProjectID*, AssignDate, StaffID*, Hours, Remarks)

3NF

CategoryID → CategoryName (Transitive dependency)

JobID → JobName (Transitive dependency)

Project (ProjectID , ProjectName, (categoryID*)

Category (CategoryID , (categoryName)

Staff (StaffID , StaffName, JobID*)

Job (JobID , JobName)

ProjectAssign (ProjectID* , AssignDate , StaffID* , Hours, Remarks)

Question 3 b)

Insertion Anomaly:

It is not possible to insert new record of project in the ProjectAssign table unless the project has been assigned to a staff

Modification Anomaly:

When we update the ProjectName of the record 'Restaurant A' (P3322) from 'Restaurant A' to 'Restaurant AB', we also have to update the ProjectName value of other similar or associated rows.

Deletion Anomaly:

When we delete the record 'Slot Game' (P3888) from ProjectAssign table, the staff record 'Kelvin' (S3388) will also be deleted.

BACS3183 ADVANCED DATABASE MANAGEMENT**Question 4**

- a) Based on the following set of WorkerID for the Worker table:

WorkerID	WorkerName	WorkerGender	Department
112	Michelle Lim	F	Assembly
127	Steven Ong	M	Control Check
130	Eve Tang	F	Control Check
132	Eric Heng	M	Assembly
148	Amy Loh	F	Assembly
153	Alice Lee	F	Material Purchasing

Table 2: Worker Table

- (i) Construct a *B+-tree* final structure of order 3 (6 marks)
- (ii) Construct a *B+-tree* final structure of order 4 (3 marks)
- b) Use the transactions T1 and T2 below to illustrate and explain the lost update and uncommitted dependency problems:

T1	T2
begin-trans	begin-trans
read (A)	read (A)
read (B)	$A = A * 20$
$A = A - B$	write (A)
write (A)	read (B)
	$B = B + 10$
	write (B)

Assuming that the initial values of A and B are 100 and 50 respectively, show clearly how each of the problems would happen when T1 starts before T2. Meanwhile each transaction can be either commit or rollback after its value is written.

- (i) Lost Update (5 + 3 marks)
- (ii) Uncommitted Dependency (5 + 3 marks)

[Total: 25 marks]

Question 4 a) (i)

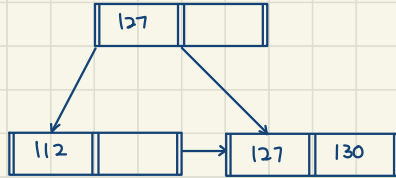
Step 1



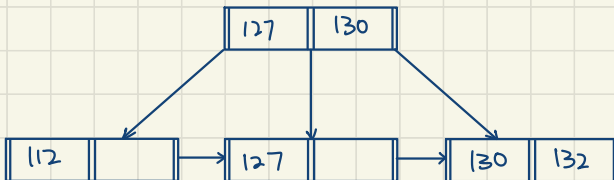
Step 2



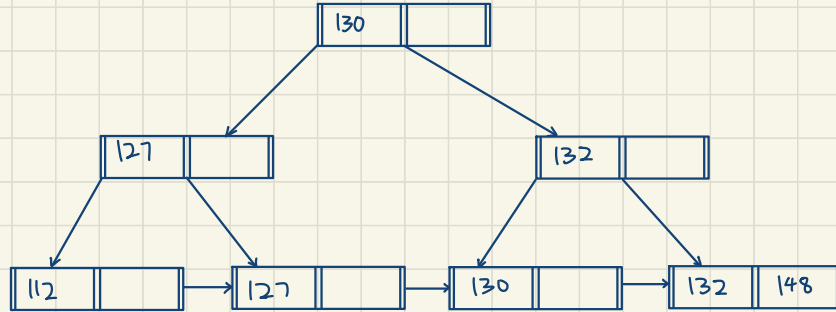
Step 3



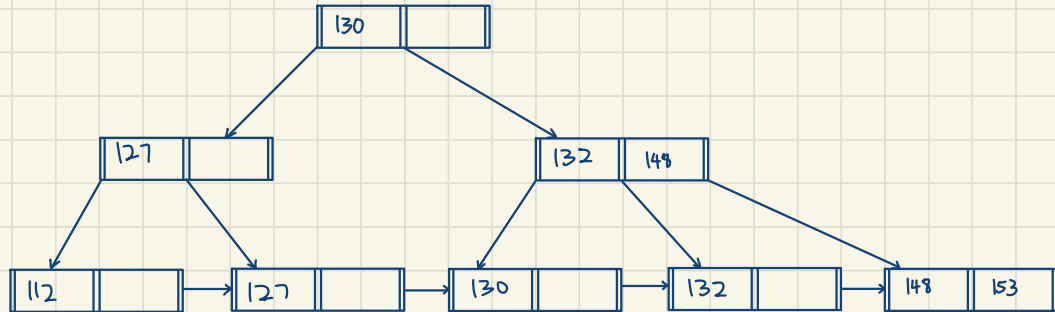
Step 4



Step 5



Step 6

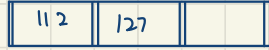


Question 4 a) (i)

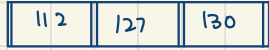
Step 1



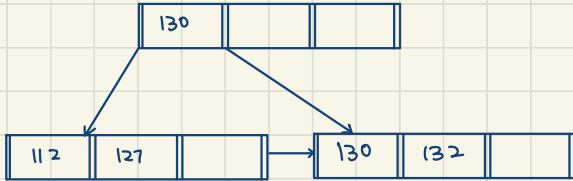
Step 2



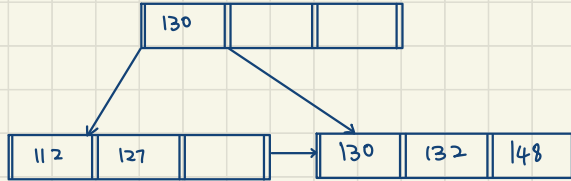
Step 3



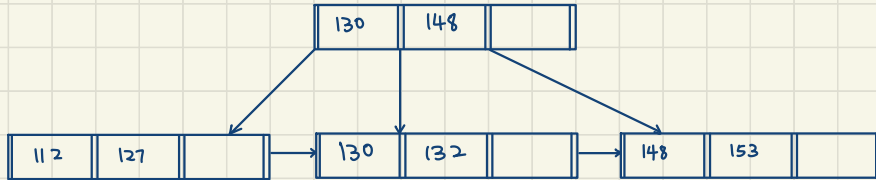
Step 4



Step 5



Step 6



Question 4 b) (i)

- Lost Update may happen when T1 starts before T2. However, T1 and T2 will read the same values since T1 has not update the data item yet.
- Then, T1 updates the data item with a new value 1. Then, the T2 updates the same data item with another new value 2.
- Since T2 did not get the latest updated value 1 from the data item before updating, T2 transaction will overwrite the T1 transaction by referring the initial value of data item before any updating.
- The final data item value will be the value 2 without considering T1 transaction.

Time	T1	T2	A	B
t_1	begin-trans		100	50
t_2	read(A)		100	50
t_3	read(B)	begin-trans	100	50
t_4	$A = A - B$	read(A)	100	50
t_5	write(A)	$A = A * 20$	50	50
t_6	commit	write(A)	1000	50
t_7		read(B)	1000	50
t_8		$B = B + 10$	1000	50
t_9		write(B)	1000	60
t_{10}		commit	1000	60

Question 4 b) (ii)

- Uncommitted dependency may happen when one transaction can see intermediate results of another transaction before it has committed.
- Since T1 starts first, T2 may be able to read the intermediate updated value of A before T1 committed the transaction.
- Thus, T2 will take the uncertain value of A to do its subsequent transaction. If there is a new updates from T1 transaction, the result of T2 transaction will be incorrect and causes the value of A inconsistent.

Time	T1	T2	A	B
t_1	begin-trans		100	50
t_2	read(A)		100	50
t_3	read(B)		100	50
t_4	$A = A - B$		100	50
t_5	write(A)	begin-trans	50	50
t_6	:	read(A)	50	50
t_7	rollback	$A = A * 20$	100	50
t_8		write(A)	1000	50
t_9		read(B)	1000	50
t_{10}		$B = B + 10$	1000	50
t_{11}		write(B)	1000	60
t_{12}		commit	1000	60