SAMPLE EXAM ONLY. DO NOT DISTRIBUTE UNTIL WEEK 12.

Monash University

FIT3171

SAMPLE EXAM PAPER S1 2019

- **Page 1 is normally reserved for the Exams Coversheet.**

- **IMPORTANT NOTE: This Sample Exam serves to provide a general overview of the general structure of the exam paper only.**

- **To protect the integrity of the exam:**
NO ACTUAL EXAM QUESTIONS are included; and the COMPOSITION OF THE SUBQUESTIONS are SUBJECT TO CHANGE.

- **Students are reminded that all content specified by the Unit Guide is examinable, including but not limited to Pre-reading (weekly Coronel & Morris chapters) + Lecture Notes + Tute Notes + all other Moodle Material (except where explicitly stated).**

Note before beginning:

You are required to write all answers in the provided script book.
ANYTHING WRITTEN IN THIS EXAM QUESTION BOOKLET IS NOT MARKED.

(Best wishes and good luck, from all of us at Unit Management).

Q1 The Relational Model (Total=10 marks)

(1a) ATTRIBUTES

A company wishes to record the following attributes about their employees: employees ID, department number, name, home address, education qualifications and skills which the employee has. A small sample of data is show below:

Employee ID	Department Number	Employee Name	Home Address	Qualification	Skill
101	21	Given name: Joe Family name: Bloggs	Street: 12 Wide Rd Town: Mytown Postcode: 1234	-Bachelor of Commerce -MBA	-Project Management -Hadoop -R
102	13	Given name: Wendy Family name: Xiu	Street: 55 Narrow St Town: Mytown Postcode: 1234	-Bachelor of Computer Science -Master of IT Doctor of Philosophy	-SQL -PL/SQL
103	13	Given name: Sarah Family name: Green	Street: 25 High St Rd Town: Mytown Postcode: 1234	-Certificate IV in Business Administration	-SQL -Java -Python

Using ONLY the data from the sample above, explain the difference between a multivalued attribute and a composite attribute. Justify your answer with CORRECT examples.

[4 marks]

(1a) REFER FIT2094 SAMPLE ANSWERS

(1b) RELATIONAL ALGEBRA

International conservation programmes include cooperation between zoos around the world to maintain and breed populations of animals, especially species that may become threatened in the wild. In this programme, different zoos share data about animals and about the health problems they encounter in keeping them.

The database contains basic data about zoos, about the species represented, about individual animals, and about the veterinary surgeons who tend to their health. A record is kept of ailments suffered by each animal and how they were treated.

Taxon

taxon_id	char(10)	NN	PK
genus	varchar2(30)	NN	
species	varchar2(30)	NN	
description	varchar2(1000)		
ref_url	char(17)		

Zoo

zoo_name	varchar2(30)	NN	PK
country	varchar2(30)	NN	
city	varchar2(30)	NN	
zoo_phone	number(12)		
zoo_address	varchar2(30)	NN	

Animal

animal_id	number(4)	NN	PK
taxon_id	char(10)	NN	FK
zoo_name	varchar2(30)	NN	FK
dob	date		
sex	char(1)	NN	

Vet

vet_name	varchar2(30)	NN	PK
vet_phone	number(12)		NN
vet_address	varchar2(30)	NN	
vet_country	varchar2(30)	NN	
vet_city	varchar2(30)	NN	

Ailment

case_date	date	NN	PK
animal_id	number(4)	NN	PFK
vet_name	varchar2(30)	NN	FK
condition	varchar2(30)	NN	
description	varchar2(1000)		

Please Note: NN = Not Null, PK = Primary key, FK = Foreign key

*** CONTINUED NEXT PAGE ***

With reference to the description of the Zoo animal care database above, write Relational Algebra statements, using symbolic notation, to produce the following information.

These rules MUST be followed when writing your algebra:

- you MUST use symbolic notation,
- if the relational algebra is complex and contains several operations, you may wish to split your answer into several steps, such as:
  R1 <- relational algebra expression-1
  R2 <- relational algebra expression-2 that uses R1 as an input, etc,
- and, where a query has several possible solutions, your relational algebra must represent an EFFICIENT solution.

(i) List the names and phone numbers of all vets **[2 marks]**
(ii) List the animal id and zoo name for all female 'F'' animals in the system **[2 marks]**
(iii) List the genus and species of all animals from South African zoos that have received treatment. **[2 marks]**

*** Q1 ENDS HERE ***

(1b) REFER FIT2094 SAMPLE ANSWERS

Q2. Database Design (Total=20 marks)

Case study below needs to be understood before commencing.

The Charity Football League - a fundraising under-sixteen football league for local communities, whose format is modelled after the English Premier League - needs a database to help track teams, children that sign up to play in the league, the parents of these football players and the coaches for each team.

The league wishes to record the details for each parent of a particular player (the parents last name, first name, phone contact number and address).

For each player the system needs to record the player's last name, first name, blood type and their date of birth. Any allergies that the player has also need to be recorded. A player may only play in one team with this league.

Each team is given a unique id: the system needs to record this id, the team's name and the city their home ground is located in (a city may have several under-sixteen teams based in it - e.g. Manchester can have Manchester Victory or Manchester Rovers).

A team's colours are also to be recorded, teams may choose to use a single or multiple team colours. A team may have several coaches – one of the coaches is designated as the head coach. A coach is only permitted to coach one team in the under-sixteen league. All communications from the league to the team are via the head coach. To be registered to play in this league a team must have a head coach and at least one player. The database needs to track a coach's first and last name, phone and address contact details and the team that they are coaching.

Create ***a logical level diagram using Crow's foot notations*** to represent the "Charity Football League" data requirements described above. Clearly state any assumptions you make when creating the model. Be sure to include all attributes. Identify clearly the Primary Keys and Foreign Keys, as part of your design. You do not need to include the data type of the attributes.

*** Q2 ENDS HERE ***

(2) REFER FIT2094 SAMPLE ANSWERS

Q3. Normalisation (Total=20 marks).

Refer to the following business case and Report.

Monash University owns several performance halls that are used by organizations within and outside Monash University.

The **following report** shows the booking information for several performances across different venues in Monash University. Multiple performances or shows can be organized at the same time across multiple venues. **For consistency, we call performances/shows as events.**

The following situations are observed during the operation of the performance halls:

- Each show is organised by an organisation.
- A single contact number is kept for an organiser.
- A show can be scheduled on multiple days, for example Melbourne Symphony-Summer Classic has two performances on the 12-Jan-2019 and 17-Jan-2019.
- A show can be scheduled twice a day (matinee and night).

Start Time	Date	Venue	Venue Location	Show Type	Show	Organiser	Organiser's contact
8 PM	12-Jan-2019	Robert Blackwood Hall	Clayton	Music concert	Melbourne Symphony – Summer Classic	MSO	(03) 99021212
8 PM	12-Jan-2019	K3.24	Caulfield	Comedy	Adam Hill	Melbourne Comedy Festival	(03) 99031456
2 PM	14-Jan-2019	Robert Blackwood Hall	Clayton	Musical	Cats	Monash Student Association	(03) 99012233
8 PM	14-Jan-2019	Alexander Theatre	Clayton	Comedy	Dave Hughes	Melbourne Comedy Festival	(03) 99031456
8 PM	16-Jan-2019	Robert Blackwood Hall	Clayton	Music concert	Hoodoo Gurus	Mushroom Promoter	(02) 90021002
8 PM	17-Jan-2019	Robert Blackwood Hall	Clayton	Music concert	Melbourne Symphony – Summer Classic	MSO	(03) 99021212

From UNF, convert the table shown above to Third Normal Form (3NF), showing each stage of the process. Clearly state any assumptions that you make. For the report only, show

ALL dependencies via required dependency diagrams. All attributes must be included, no surrogate keys may be added during the normalisation.

*** Q3 ENDS HERE *** **(3) REFER FIT2094 SAMPLE ANSWERS**

Q4. Transaction Management (Total=10 marks)

(4a)

Using an example, illustrate and explain what the lost update problem is where two concurrent transactions are updating the same data element. Then, define the term ATOMICITY and explain how it can mitigate the problem above. **[7 marks]**

(4b)

Given the following hypothetical scenario where multiple users are accessing a slow database each with their own transactions.
Assume T1 = user anne; T2 = user bruce; T3 = user charles; T4 = user david

- anne acquires a shared lock for record A then reads it.
- bruce acquires a shared lock for record B then reads it.
- charles wants to write to record A, but anne has not finished.
- david acquires an exclusive lock for record C then changes it.
- emma acquires a shared lock for record A then reads it.
- david wants to write to record A, but anne has not finished.

Illustrate the scenario above using a Wait-For Graph (WFG), with the correct notation (T1, T2... etc) instead of usernames.

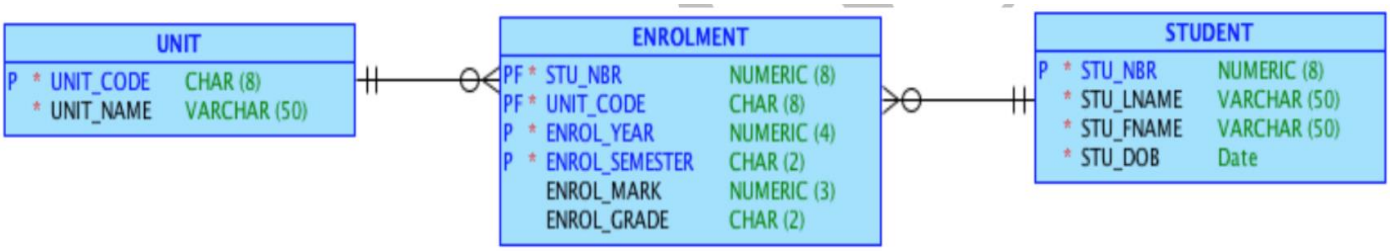
Then, explain the presence/absence of any deadlock.
You must justify your answer based on the WFG. **[3 marks]**

*** Q4 ENDS HERE ***

(4a) REFER FIT2094 SAMPLE ANSWERS
(4b) REFER FIT2094 SAMPLE ANSWERS

Q5. FIT3171: Advanced SQL, Database Theory and Implementation (Total = 40 marks)

For questions 5(a), 5(b), refer to the ENROLMENT data model below, modified from case studies found in the unit.



(5a) SQL (DQL) SELECT [10 marks]

For the ENROLMENT data model, construct SQL SELECT statements for the following.

i. Select all student names for all the students who are enrolled in FIT units (prefix of unitcode is exactly 'FIT') BUT NOT enrolled in BIZ units (prefix of unitcode is 'BIZ'). Remember to remove duplicates and use appropriate set operations. You also need to use aliases to describe the columns more sensibly, as FIRSTNAME, and LASTNAME.

ii. Create a VIEW which is made up of the joining of UNIT and ENROLMENT using table aliases, and selects all unit codes, unit names, grades, and years. In this VIEW, any grades of NULL should be substituted with 'NA', and a column alias FINAL_GRADE assigned. Then, you MUST use this newly-created VIEW to generate a summary of all grades sorted in descending order of the totals.

HINT: This is an example of what we expect for a naive example of 10 enrolments.

```
HD      5
C       2
N       2
NA      1
P       0
D       0
```

(5a) SAMPLE ANSWERS:

```
-- Part i
SELECT DISTINCT stu_lname as LASTNAME, stu_fname as FIRSTNAME
FROM enrolment e JOIN student s ON e.stu_nbr = s.stu_nbr
WHERE e.unit_code LIKE 'FIT%'
MINUS
SELECT DISTINCT stu_lname as LASTNAME, stu_fname as FIRSTNAME
FROM enrolment e JOIN student s ON e.stu_nbr = s.stu_nbr
WHERE e.unit_code LIKE 'BIZ%';

-- Part ii
CREATE VIEW UNIT_ENROLMENT_VIEW AS
    SELECT u.unit_code, u.unit_name, NVL(e.enrol_grade, 'NA') as
FINAL_GRADE, e.enrol_year
    FROM unit u JOIN enrolment e
    ON u.unit_code = e.unit_code;

SELECT final_grade, count(*)
FROM UNIT_ENROLMENT_VIEW
GROUP BY final_grade
ORDER BY count(*) DESC;
```

(5b) SQL DML [10 marks]

Refer to the ENROLMENT data model, and assume ALL the tables have ALL been created correctly in Oracle. Also assume the order of attributes the programmer used in CREATE TABLE statements follow the order it appears in the data model, top-to-bottom (e.g. for UNIT, UNIT_CODE is first, then UNIT_NAME).

i. Add an enrolment with the following details:

- student whose ID needs to be looked up, with DOB = 25 Dec 2000 and surname Smith;
- unit whose code needs to be looked up, with the exact unit name 'Graphics';
- first semester; and
- year needs to be extracted from today's date and converted to characters appropriately.
- no grades nor marks as of now.

ii. What two SQL errors and one dangerous practice are present in the SQL statement below (verbatim)?

```
UPDATE enrolment VALUES enrol_mark = '55';
```

Explain EXACTLY and provide concise justifications and corrections based on the UNIVERSITY data model.

```
(5b) SAMPLE ANSWERS:

-- Part i
-- REFER FIT2094 SAMPLE ANSWERS


-- Part ii
SQL error - UPDATE .. SET (instead of UPDATE .. VALUES)
This is wrong syntax.

SQL error - value '55'
Check the schema that enrol_mark is a numeric value, not char.
'55' specifies that it's a char!
Fix it by removing the single quotes.

Dangerous practice - WHERE .. not mentioned.
If WHERE .. is not specified, it updates everything in the table.
```

(5c) PL/SQL Triggers [10 marks]

Given the following two Oracle tables (**rem** precedes a remark which won't be executed).

```
rem
rem Table : DEPARTMENT
rem dept_no : Department Number
rem dept_name : Name of department
rem dept_total_salary : Total Salary of employees in department
rem

create table department (
   dept_no            number (2)      not null,
   dept_name          char (20)       not null,
   dept_total_salary  number (10,2)   not null,
   constraint pk_department primary key (dept_no)
);

rem Create sequence to use for department PK

create sequence dept_no_seq increment by 1 start with 100;

rem
rem Table : EMPLOYEE
rem emp_no : Employee number
rem emp_name : Employee Name
rem emp_salary : Employee Salary
rem emp_sdate : Employee start date
rem dept_no : Department Number (Foreign Key)
rem

create table employee (
   emp_no       number (4)      not null,
   emp_name     char (20)       not null,
   emp_salary   number (8,2)    not null,
   emp_sdate    date            not null,
   dept_no      number (2)      not null,
);

constraint pk_employee primary key (emp_no),
constraint fk_employee foreign key (dept_no)
references department(dept_no));
```

Write the code for an Oracle structure (or structures), that will automatically maintain the derived column dept_total_salary (the total salary for all the employees currently in the department) when modifications (insert, update or delete) are made to the employee table.

(5c) SAMPLE ANSWERS:

```
create or replace trigger total_sal_maintain
after INSERT or DELETE or UPDATE on employee
FOR each row

BEGIN
if inserting then
      update department
            set dept_total_salary = dept_total_salary + :new.emp_salary
      where
            dept_no = :new.dept_no;

else
      if deleting then
            update department
                  set dept_total_salary = dept_total_salary - :old.emp_salary
            where
                  dept_no = :old.dept_no;
      else
            if (updating ('emp_salary') or updating ('dept_no')) then
                  update department
                        set dept_total_salary = dept_total_salary
- :old.emp_salary
                  where
                        dept_no = :old.dept_no;
                  update department
                        set dept_total_salary = dept_total_salary
+ :new.emp_salary
                  where
                        dept_no = :new.dept_no;
            end if;

      end if;
end if;
END;
```

(5d) ADVANCED Database Theory and Implementation [6+4 = 10 marks]

Provide short answers to the following questions.
You are not expected to code anything in (5d).

i. Given the following description for an entity in a phone app:

A Contact has a first name, a middle initial, a last name, from zero up to three telephone numbers, a counter of incoming calls from the contact, and a counter of outgoing calls have been made to the contact. The first name, middle initial, and last name form the compound PK. Both counters are derived attributes.

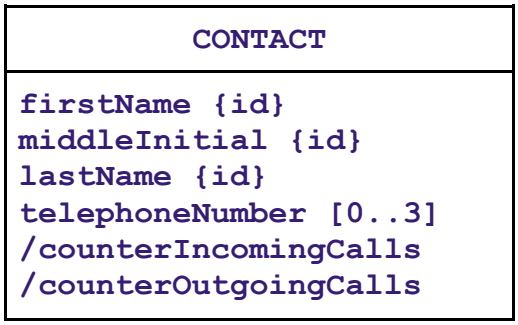
Draw the above entity using correct UML Class notation as seen in this unit.

ii. Consider the implementation of databases in a large multinational company.
Distinguish between Operational Data vs. Decision Support Data, from the following four characteristics (as per Coronel & Morris):

- Timespan
- Granularity
- Data Volume
- Data Models (structure and type of DBMS).

*** Q5 ENDS HERE ***
*** End of sample exam ***

(5d) SAMPLE ANSWERS:
-- Part i



-- Part ii
Refer Coronel and Morris Table 13.5 (in Lecture 12)

NB: Timespan refers to the period of time covered by Coronel & Morris as the characteristic 'Data Currency'.