INFO20003 Mid-Semester Test SAMPLE

School of Computing and Information Systems The University of Melbourne

Subject Number: INFO20003

Subject Title: Database Systems

Reading + Writing Time: 40 minutes

Number of Questions/Grading Scheme: 2 questions, a total of **10** marks.

This forms 10% of your overall

grade for the subject.

Authorised Materials: None

Write your student id here:	

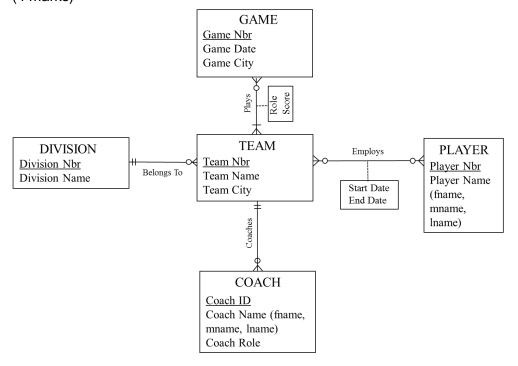
Use only the given space for each question to write down your answers

You may (and probably should) use PENCIL to answer this test!

Question 1: Database Design

A sports league wants to maintain data about a set of teams that play in the league and the players within the teams. Each player can at one point in time play only for one team but, naturally, each team has multiple players. Players move frequently from one team to another and therefore, the league needs to know when a player started to play for a particular team and when their contract with that team ended or is scheduled to end. Teams have multiple coaches, each of whom has a specific role within the team. Each team belong to a division. The league has to also keep track of the games that the teams play; relevant data about a game include its date, city it is played in, home team/visitor roles, and the score. Make sure that each entity has at least two attributes. State any assumptions deemed appropriate.

Q1-A) Draw a **Conceptual** Model (ER / EER) for the scenario described above. Write comments and reasoning in the space provided on the next page. (4 marks)



Question 2: SQL and Relational Algebra

1. Get slotMachine numbers for slotMachines played at any casino by a player from the same city as that casino.

SELECT DISTINCT Ply.SlotNum AS SlotMachine FROM Plays AS Ply, Player AS Pl, Casino AS Ca WHERE Pl.PlayerNum=Ply.PlayerNum AND Ply.CasinoNum = Ca.CasinoNum AND Pl.Hometown=Ca.Location:

SELECT DISTINCT Ply.SlotNum AS SlotMachine FROM Plays AS Ply NATURAL JOIN Player NATURAL JOIN Casino WHERE Hometown=Location;

SELECT DISTINCT Plays.SlotNum AS SlotMachine FROM Plays INNER JOIN Player ON Plays.PlayerNum = Player.PlayerNum INNER JOIN Casino ON Plays.CasinoNum = Casino.CasinoNum WHERE Hometown=Location:

Relational algebra:

$$\pi_{SlotNum}(\sigma_{hometown=location}(Player \bigvee Plays \bigvee Casino)$$

2. Name the casinos that have been visited by Sheldon.

SELECT DISTINCT Ca.CasinoName FROM Casino AS Ca, Plays AS Ply, Player AS Pl WHERE Ca.CasinoNum=Ply.CasinoNum AND Ply.PlayerNum=Pl.PlayerNum AND Pl.PlayerName = 'Sheldon';

SELECT DISTINCT CasinoName FROM Casino NATURAL JOIN Plays NATURAL JOIN Player WHERE PlayerName = 'Sheldon';

Again we can have all three possibilities like above (expressed with natural join, inner join or cross product)

Relational algebra:

 $\pi_{CasinoName}(\sigma_{Playername="Sheldon"}(Casino \bowtie Plays \bowtie Player)$

3. Get the total amount of totalBets per slotMachine for player P50.

SELECT SUM(TotalBet) AS TotalAmount, slotNum as Slot FROM Plays
WHERE PlayerNum='P50'
GROUP BY slotNum;

4. Name the slotMachines located in any casino in LasVegas.

SELECT DISTINCT SI.SlotName AS SlotMachines FROM Plays AS ply, Slots AS SI, Casino AS Ca WHERE SI.SlotNum=Ply.SlotNum
AND Ply.CasinoNum=Ca.CasinoNum
AND Ca.Location='LasVegas';

Again we can have all three possibilities like above (expressed with natural join, inner join or cross product)

5. Print all player numbers for players who are at least Level 20 and have bet more than 1000.

SELECT PI.PlayerNum
FROM Plays AS Ply, Player AS Pl
WHERE PI.PlayerNum=Ply.PlayerNum
AND Level >=20
GROUP BY PI.PlayerNum
HAVING SUM(TotalBet) > 1000;