	Q1	Q2	Q3	Q4	Total
Max Points	**20**	**20**	**35 (5 each query)**	**30 (5 each sub-question)**	**105**

CS3431 (Database Systems I)
Midterm Exam
B-term, 2011
(90 Minutes)

Student Name:

WPI ID:

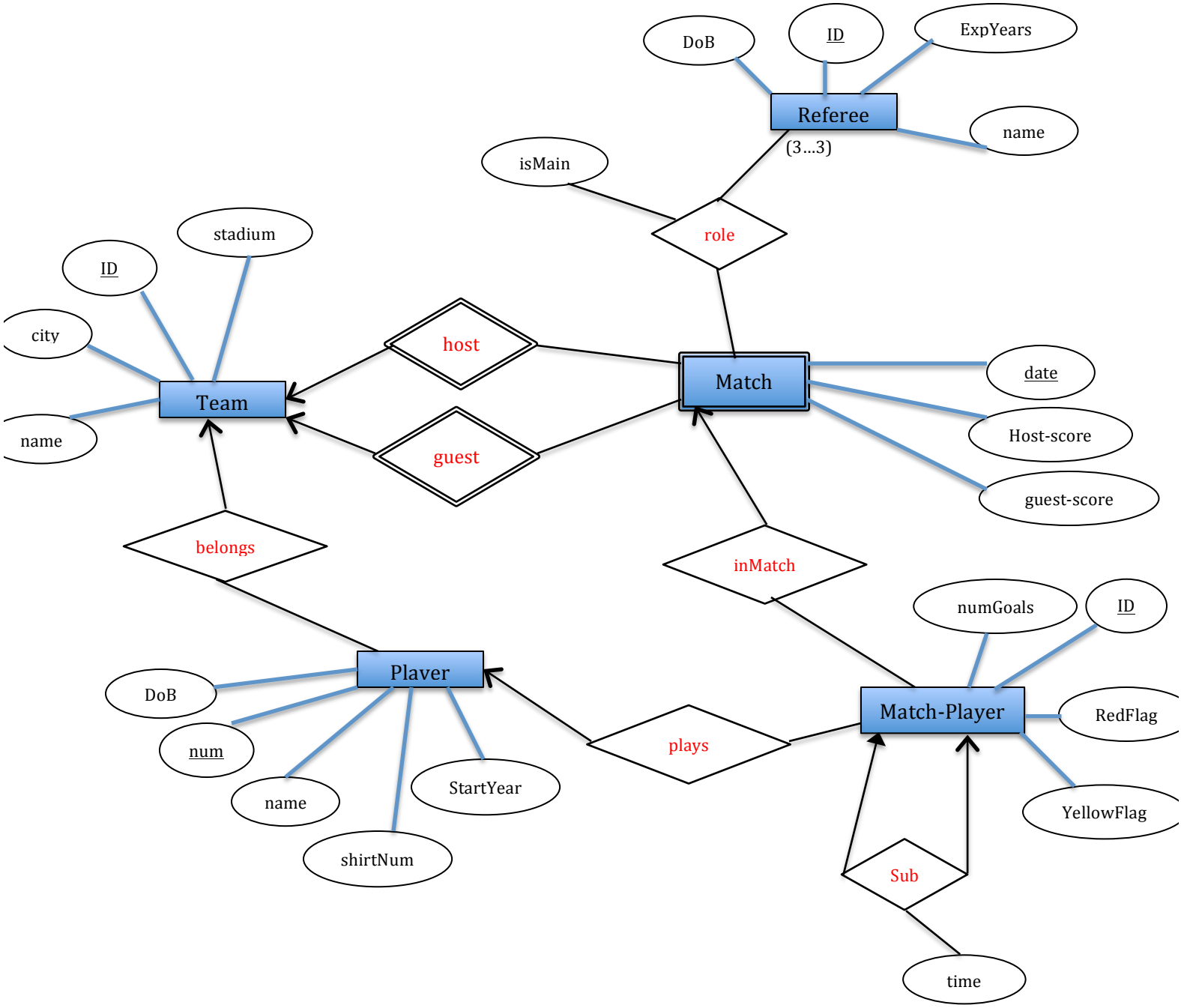
Total 10 Pages. Answer in the same sheet.

Question 1 (Design ER Diagram):

Assume we have the following application that models soccer teams, the games they play, and the players in each team. In the design, we want to capture the following:

- We have a set of teams, each team has an ID (unique identifier), name, main stadium, and to which city this team belongs.
- Each team has many players, and each player belongs to one team. Each player has a number (unique identifier), name, DoB, start year, and shirt number that he uses.
- Teams play matches, in each match there is a host team and a guest team. The match takes place in the stadium of the host team.
- For each match we need to keep track of the following:
    - The date on which the game is played
    - The final result of the match
    - The players participated in the match. For each player, how many goals he scored, whether or not he took yellow card, and whether or not he took red card.
    - During the match, one player may substitute another player. We want to capture this substitution and the time at which it took place.
- Each match has exactly three referees. For each referee we have an ID (unique identifier), name, DoB, years of experience. One referee is the main referee and the other two are assistant referee.

Design an ER diagram to capture the above requirements. State any assumptions you have that affects your design (use the back of the page if needed). Make sure cardinalities and primary keys are clear.



Assumptions:

1- In Match-Player entity set, we added a unique identifier for each record ID.
2- The final result in Match entity set is captured using two attributes Host-score and guest-score
3- The attribute 'isMain' in relationship 'role' is true if the referee is the main referee in the match, otherwise, it will be false.

Question 2 (Relational Model):

Map the ERD in Question 1 to create the relational model corresponding to the described application. Basically, list the CREATE TABLE statements with the attribute names, and appropriate data types. Also make sure to have the primary keys and foreign keys clearly defined (use the back of the page if needed).

```
Create Table Team (
 ID: int Primary Key,
City: varchar(100),
Name: varchar(100),
Stadium: varchar(100));
```

```
Create Table Player (
 num: int Primary Key,
DoB: date,
Name: varchar(100),
StartYear: int,
ShirtNum: int,
TeamID: int Foreign Key References Team(ID));
```

```
Create Table Referee (
 ID: int Primary Key,
DoB: date,
Name: varchar(100),
ExpYear: int);
```

```
Create Table Match (
 HostID: int Foreign Key References Team(ID),
GuestID: int Foreign Key References Team(ID),
Date: date,
Host-score: int,
Guest-score: int,
Primary Key (HostID, GuestID, Date));
```

```
Create Table RefereeRole (
 HostID: int,
GuestID: int,
Date: date,
RefID: int Foreign Key References Referee(ID),
isMain: Boolean,
Foreign Key (HostID, GuestID, Date) References Match (HostID, GuestID, Date),
Primary Key (HostID, GuestID, Date, RefID);
```

```
Create Table Match-Player (
 ID: int Primary Key,
PlayerNum: int Foreign Key References Player(num),
MatchDate: date,
HostID: int,
GuestID: int,
numGoals: int,
redFlag: Boolean,
yellowFlag: Boolean,
subID: int Foreign Key References Match-Player(ID),
subTime: int,
Foreign Key (HostID, GuestID, MatchDate) References Match (HostID, GuestID, Date));
```

Question 3 (Relational Algebra):

Consider the following relations:

Doctor(SSN, FirstName, LastName, Specialty, YearsOfExperience, PhoneNum)
Patient(SSN, FirstName, LastName, Address, DOB, PrimaryDoctor_SSN)
Medicine(TradeName, UnitPrice, GenericFlag)
Prescription(Id, Date, Doctor_SSN, Patient_SSN)
Prescription_Medicine(Prescription Id, TradeName, NumOfUnits)

- The **Doctor** relation has attributes Social Security Number (SSN), first and last names, specialty, the number of experience years, and the phone number.
- The **Patient** relation has attributes SSN, first and last names, address, date of birth (DOB), and the SSN of the patient's primary doctor.
- The **Medicine** relation has attributes trade name, unit price, and whether or not the medicine is generic (True or False).
- The **Prescription** relation has attributes the prescription id, the date in which the prescription is written, the SSN of the doctor who wrote the prescription, and the SSN of the patient to whom the prescription is written.
- The **Prescription_Medicine** relation stores the medicines written in each prescription along with their quantities (number of units).

Write the relational algebra expressions for the following queries (consider the three performance/optimization rules taken in class)

1. **List the trade name of generic medicine with unit price less than $50.**

$$\Pi_{\text{TradeName}}\left(\sigma_{\text{genereicFlag=True and UnitPrice} < 50}(\text{Medicine})\right)$$

2. List the first and last name of patients whose primary doctor named 'John Smith'.

$R1 \leftarrow \Pi_{SSN}(\sigma_{FirstName = 'John' \text{ and } LastName = 'Smith'}(Doctor))$

$Result \leftarrow \Pi_{FirstName,\ LastName}(R1 \bowtie_{SSN=PrimaryDoctor\_SSN}(Patient))$

3. List the first and last name of doctors who are not primary doctors to any patient.

$R1 \leftarrow \Pi_{SSN}(Doctor) - \Pi_{SSN \leftarrow PrimaryDoctor\_SSN}(Patient)$

$Result \leftarrow \Pi_{FirstName,\ LastName}(R1 \bowtie Doctor)$

4. **For medicines written in more than 20 prescriptions, report the trade name and the total number of units prescribed.**

$$R1 \leftarrow \gamma_{TradeName,\ CNT \leftarrow count(Prescription\_Id),\ SUM \leftarrow sum(NumOfUnits)}\ (Prescription\_Medicine))$$

$$Result \leftarrow \pi_{TradeName,\ SUM}\ (\sigma_{CNT > 20}\ (R1))$$

5. **List the SSN of patients who have 'Aspirin' and 'Vitamin' trade names in one prescription.**

$$R1 \leftarrow \pi_{Id \leftarrow PM1.Prescription\_Id}\ (\rho_{PM1}(Prescription\_Medicine) \bowtie_{PM1.Prescription\_Id = PM2.Prescription\_Id\ AND\ PM1.TradeName='Aspirin'\ AND\ PM2.TradeName='Vitamin'} \rho_{PM2}(Prescription\_Medicine))$$

$$Result \leftarrow \pi_{Patient\_SSN}\ (R1 \bowtie Prescription)$$

6. **List the SNN of distinct patients who have 'Aspirin' prescribed to them by doctor named 'John Smith'.**

$R1 \leftarrow \pi_{ID}(\sigma_{FirstName = \text{'John' AND } LastName=\text{'Smith'}}(Doctor) \bowtie_{SSN=Doctor\_SSN} Prescription)$

$R2 \leftarrow \pi_{ID \leftarrow Prescription\_id}(\sigma_{TradeName=\text{'Aspirin'}}(Prescription\_Medicine)) \cap R1$

$Result \leftarrow \delta(\pi_{Pateint\_SSN}(R2 \bowtie Prescription))$

7. **List the first and last name of patients who have no prescriptions written by doctors other than their primary doctors.**

$R1 \leftarrow \pi_{SSN}(Patient \bowtie_{SSN=Patient\_SSN \text{ AND } Doctor\_SSN<>PrimaryDoctor\_SSN} Prescription)$

$R2 \leftarrow \pi_{SSN}(Patient) - R1$

$Result \leftarrow \pi_{FirstName,\ LastName}(R2 \bowtie Patient)$

Question 4 (Functional Dependencies and Normalization):

Given the following relation R = (A, B, C, D, E, F), and the following dependencies
F = {AB → DE, CD → E, B → EF, DF → AC, BD → AF}

4.1) Report the candidate keys of R

{AB} , where $\{AB\}^+ = \{ABCDEF\}$ and none of A^+ or B^+ contains all keys

{BD} , where $\{BD\}^+ = \{ABCDEF\}$ and none of D^+ or B^+ contains all keys

4.2) Report a canonical cover for R (the minimal subset of FDs that has the same functional closure) (use the back of the page if needed)

Canonical cover is:
G = {AB → D, CD → E, B → EF, DF → AC}

4.3) Which of the five given FDs violates the BCNF (if any)?

CD → E,
B → EF,
DF → AC, and

4.4) If R is not in BCNF, provide decomposition into multiple relations where each one becomes in BCNF. For each decomposition step, clearly identify which FD you use for the decomposition (use the back of the page if needed).

Using (CD → E) to divide R
R1 = (C, D, E), R2 = (A, B, C, D, F)

R2 is still not in BCNF and violated by: B→ F and DF → AC

Using (B → F) to divide R2
R1 = (C, D, E), R3 = (B, F), R4 = (A, B, C, D)

Now R1, R3, R4 are in BCNF

4.5) What are the attribute closures of {DF} and {BC}

$\{DF\}^+ = \{ACDEF\}$
$\{BC\}^+ = \{BCEF\}$

4.6) If R is decomposed into two relations R1 = (A, B, D, F) and R2 = (C, D, E, F), is it dependency preserving or not? If not which FDs are lost?

In R1 we have:
F1 = {AB → D, B → F, DF → A, BD → AF}

In R2 we have:
F1 = {CD → E, DF → C}

The original FDs are:
AB → DE – Preserved using (AB→D, B → F) from R1 and (DF → C) from R2

CD → E -- Preserved from (CD → E) in R2

B → F -- Preserved from (B→ F) in R1

B → E -- Lost

DF → AC -- Preserved from (DF → A) from R1, and (DF → C) from R2

BD → AF -- Preserved from (BD → AF) from R1

The decomposition is not dependency preserving as B→ E is lost.