Last Name: _____   First Name: _____   Student Id: _____

Number: _____

I hereby pledge that I will stay truth to UHCL's Honor Code.

Signature: **Date:**

Time allowed: one hour 20 minutes. Total score: 100 points. *Closed* book examination.

Answer all questions. Write in the back of the question paper, if necessary. <u>Turn in any additional answer papers and rough work.</u>

Academic honesty policy will be followed strictly. Cheating will be pursued vigorously and will result in a failing grade of D or below, a permanent academic record and possibly other more serious penalty!

(1) [30 points] The goal is to build a toy prototype database to *partially support a league of teams.* Construct an UML class diagram to capture and model the partial requirements below. You should list class names, attributes with multiplicities, and associations with multiplicities. Roles of associations should also be provided when appropriate. Multiplicities should be as specific as possible. Show the stereotype <<PK>> (primary key) and <<unique>> (unique) when applicable.

The league is composed of teams. The names of teams must be stored. There are players in the league. The last name, first name, phone and email of a player should be stored. However, the email is optional. A player has a unique id. A player can join as a team member for at most one team. A team must have a player as its captain.
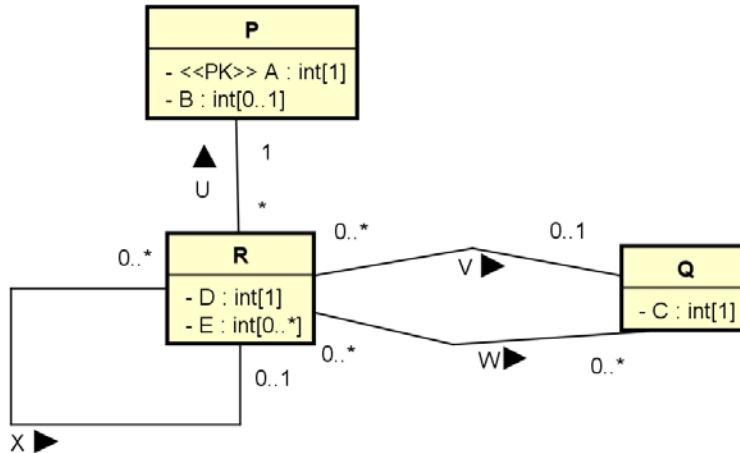
There are coaches. They must be registered to serve as coaches. CoachId is a unique identifier of a coach and it must be stored. The name and email address of a coach should also be stored. A team must have a main coach and may have up to two assistant coaches. A coach can serve for any number of teams in various capacities.

A game is conducted between two teams (team 1 and team 2). The scores of the two competing teams and the game date should be stored. A team can play in any number of games.
You may answer in the next page.

Answer for question (1):

(2) [15 points] Consider the data model in the following UML class diagram. Construct a reasonable set of relation schema to implement it by filling in the table below as necessary. For each relation, list its candidate keys **(CK)**, foreign keys **(FK)**, and all attributes you know for sure that are *not nullable* **(NN)**. Ignore data types. Provide additional notes (such as whether a surrogate key has been created) if appropriate.

P
- <<PK>> A : int[1]
- B : int[0..1]

1
U
*

0..*   0..1

0..*
R
- D : int[1]
- E : int[0..*]

V ▶

Q
- C : int[1]

0..*   W ▶   0..*

0..1
X ▶

Relation		Relation	
[CK]		[CK]	
[FK]		[FK]	
[NN]		[NN]	
[Note]		[Note]	

Relation		Relation	
[CK]		[CK]	
[FK]		[FK]	
[NN]		[NN]	
[Note]		[Note]	

Relation		Relation	
[CK]		[CK]	
[FK]		[FK]	
[NN]		[NN]	
[Note]		[Note]	

(3) [20 points]

(a) [T or F] In the relational model, a relation always has at least one superkey.

(b) [T or F] R has 3 tuples. S has 4 tuples. As a result R * S always has 12 tuples.

(c) [T or F] It is possible for R(A,B,C,D) to have four foreign keys.

(d) [T or F] The relational model is based on the concept of two dimensional arrays.

(e) [T or F] Relational calculus is an object-oriented language.

(f) [T or F] Relational algebra is an object-oriented language.

(g) [T or F] Comparing to file systems, an important advantage of relational DBMS is their better support of concurrent access.

(h) [T or F] When mapping an UML class diagrams to relational schema, an association is always mapped to a new relation.

(i) [T or F] In SQL, null is a built-in marker indicating missing information.

(j) [T or F] It is possible for R(A,B,C,D,E) to have exactly three superkeys.

(4) [5 points] It is known that for the relation R(A,B,C,D,E):

1. There are two candidate keys.
2. C and D are not a prime attribute.
3. BE is not a superkey.

Can you deduce what the two CK are? Explain.

Questions 5 and 7 use the following relations with an example of an instance shown below.

Supplier(<u>SNum</u>, SName, SCity, Status)
Part(<u>PNum</u>, PName, Color, Weight)
Supply(<u>SNum</u>, <u>PNum</u>, Quantity); FK: SNum references Supplier(SNum); PNum references Part(PNum)

Supplier:

SNum	SName	SCity	Status
S1	ABC	Dallas	10
S2	DEF	Houston	20
S3	Go go	Houston	12
S4	P&G	Dallas	2
S5	Yap	Phoenix	5
S6	Yue	Dallas	1

Part:

PNum	PName	Color	Weight
P1	Drum	Green	10
P2	Hammer	Green	20
P3	Minipod	Red	4
P4	Micropod	Red	4
P5	Blue Spur	Blue	3
P6	Musical Box	Blue	13
P7	Bear	Blue	9
P8	Panda	White	10

Supply:

SNum	PNum	Quantity
S1	P1	10
S1	P2	3
S2	P1	11
S2	P2	1
S2	P4	6
S3	P4	1
S3	P5	2
S3	P6	12
S3	P7	5
S4	P2	1
S4	P5	10
S4	P7	4
S4	P8	10
S5	P1	11
S5	P3	5
S5	P4	10
S5	P5	14

(5) [15 points] Write the relational algebra expressions for the following queries. You may use either the mathematical notation *or* the syntax of the RA interpreter in HW #3.

(a) List the Pnum, name, and weight of all parts supplied by a supplier in Dallas or Houston, and has a status smaller than 11.

```
+------+-----------+--------+
| pnum | pname     | weight |
+------+-----------+--------+
| P1   | Drum      |     10 |
| P2   | Hammer    |     20 |
| P5   | Blue Spur |      3 |
| P7   | Bear      |      9 |
| P8   | Panda     |     10 |
+------+-----------+--------+
```

(b) List the names of suppliers that do not supply any part with a weight of 10 or more.

```
+-------+
| sname |
+-------+
| Yue   |
+-------+
```

(c) List the name and status of every supplier that supplies a green part and a part with weight more than or equal to 10.

```
+-------+--------+
| sname | status |
+-------+--------+
| ABC   |     10 |
| DEF   |     20 |
| P&G   |      2 |
| Yap   |      5 |
+-------+--------+
```

(6) [10 points] Write the *DRC* expressions for the following queries of Q5.

(a) List the Pnum, name, and weight of all parts supplied by a supplier in Dallas or Houston, and has a status smaller than 11.

```
+------+-----------+--------+
| pnum | pname     | weight |
+------+-----------+--------+
| P1   | Drum      |     10 |
| P2   | Hammer    |     20 |
| P5   | Blue Spur |      3 |
| P7   | Bear      |      9 |
| P8   | Panda     |     10 |
+------+-----------+--------+
```

(b) List the names of suppliers that do not supply any part with a weight of 10 or more.

```
+-------+
| sname |
+-------+
| Yue   |
+-------+
```

(c) List the name and status of every supplier that supplies a green part and a part with weight more than or equal to 10.

```
+-------+--------+
| sname | status |
+-------+--------+
| ABC   |     10 |
| DEF   |     20 |
| P&G   |      2 |
| Yap   |      5 |
+-------+--------+
```

(7) [15 points] Write SQL statements for the following queries of Q5.

(a) List the Pnum, name, and weight of all parts supplied by a supplier in Dallas or Houston, and has a status smaller than 11.

```
+------+-----------+--------+
| pnum | pname     | weight |
+------+-----------+--------+
| P1   | Drum      |    10  |
| P2   | Hammer    |    20  |
| P5   | Blue Spur |     3  |
| P7   | Bear      |     9  |
| P8   | Panda     |    10  |
+------+-----------+--------+
```

(b) List the names of suppliers that do not supply any part with a weight of 10 or more.

```
+-------+
| sname |
+-------+
| Yue   |
+-------+
```

(c) List the name and status of every supplier that supplies a green part and a part with weight more than or equal to 10.

```
+-------+--------+
| sname | status |
+-------+--------+
| ABC   |    10  |
| DEF   |    20  |
| P&G   |     2  |
| Yap   |     5  |
+-------+--------+
```