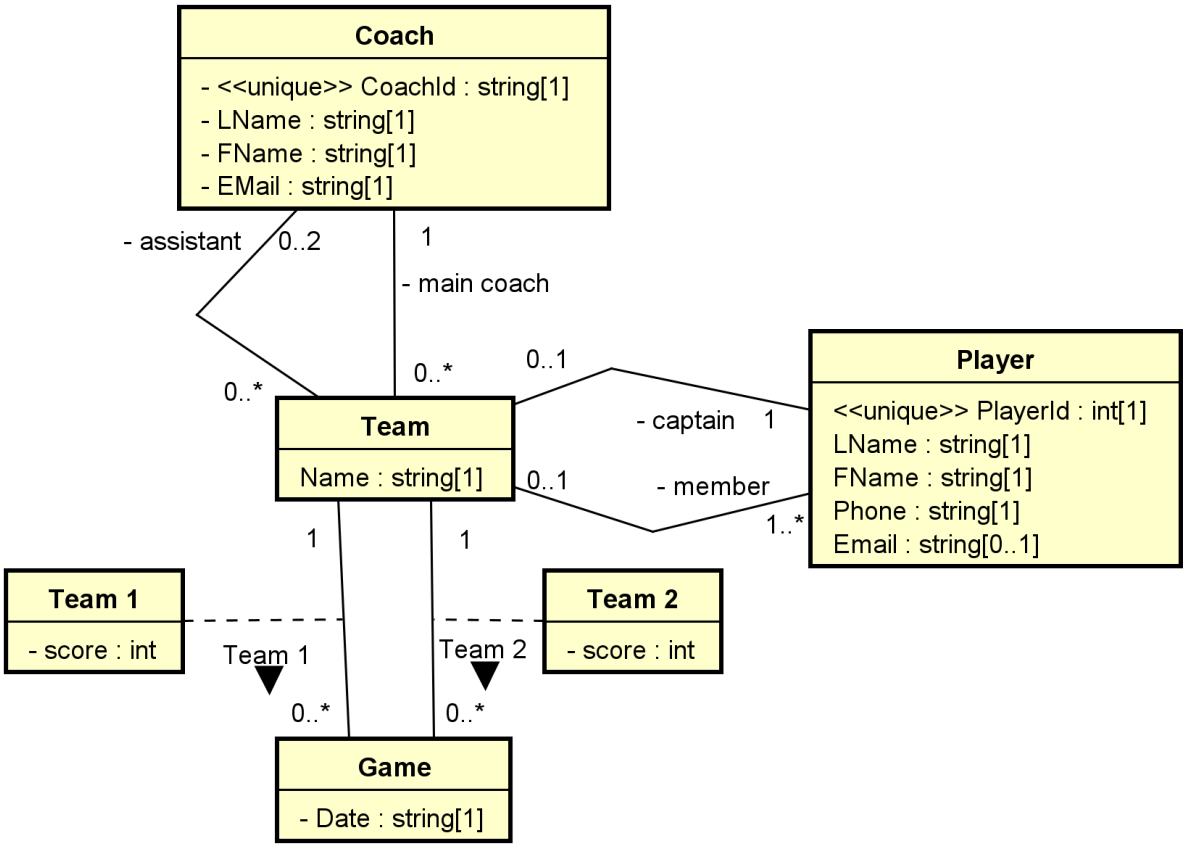
CSCI 5333 DBMS

Spring 2020

Suggested Solution to Mid-Term Examination

(1) For example (data types not required): Team 1 and Team 2 may be modeled as associations instead of association classes. In this case, the attributes, team_1_score and team_2_score should be added to the class Game.



(2) For example:

Relation	P(A, B)	**Relation**	Q(QID, C)
[CK] [1] A [FK] [NN] A [Note]		[CK] [1] QID [FK] [NN] QID, C [Note] QID is created as the surrogate primary key.	
Relation	R(RID, D, A, V_QID, X_RID)	**Relation**	W(WID, RID, QID)
[CK] [1] RID [FK] [1] A references P(A); [2] V_QID references Q(QID); [3] X_RID references R(RID) [NN] RID, D, A [Note] RID is created as the surrogate primary		[CK] [1] WID, [2] RID, QID [FK] [1] RID references R(RID); [2] QID references Q(QID) [NN] WID, RID, QID [Note] WID is created as the surrogate primary	

key.		key.	
Relation	RE(REID, RID, E)	**Relation**	
[CK] [1] REID, [2] RID, E [FK] [1[RID references R(RID) [NN] REID, RID, E [Note] REID is created as the surrogate primary key.		[CK] [FK] [NN] [Note]	

(3) (a) T (b) T (c) T (d) F (e) F
(f) F (g) T (h) F (i) T (j) T

(4) Yes, AB and AE.

[1] C and D are non-prime attributes. Thus, the CK can be made of only A, B and E. Potential choices: A, B, E, AB, AE, BE and ABE.

[2] BE is not a SK. Thus, B, E and BE cannot be CK. Remaining potential choices: A, AB, AE and ABE.

[3] There are two CK. Only AB and AE can be CK at the same time, and thus they are the two CKs.

(5) For example,

(a) $\pi_{PNum, PName, Weight}(\sigma_{(SCity='Dallas' \vee SCity='Houston') \wedge status<11}(SUPPLIER) \mid X \mid SUPPLY \mid X \mid (Part))$

```
project [pnum, pname, weight]
(((project [snum] (select [scity='Houston'] (supplier)))
 union
(project [snum] (select [scity='Dallas'] (supplier))))
join
(project [snum] (select [status<11] (supplier)))
join
supply
join
part);
```

(b) $\pi_{SName}(SUPPLIER \mid X \mid (\pi_{SNum}(SUPPLIER) - \pi_{SNum}(SUPPLY \mid X \mid \sigma_{Weight<=10}(PART)))$

```
project [sname]
(supplier
join
((project [snum] (supplier))
minus
(project [snum]
 (supply join (select [weight <= 10] (part))))));
```

(c) $\pi_{SName, Status}(SUPPLIER \mid X \mid (\pi_{SNum}(SUPPLY \mid X \mid \sigma_{Color='Green'}(PART)) \cap \pi_{SNum}(SUPPLY \mid X \mid \sigma_{Weight>=10}(PART))))$

```
project [sname, status]
(supplier
```

```
join
((project [snum] (supply join (select [color='Green'] (part))))
intersect
(project [snum] (supply join (select [weight >= 10] (part))))));
```

(6) (a)

{(pnum, pname, weight) | (snum,_,scity,status) ∈ Supplier, (snum, pnum, _) ∈ Supply, (pnum,pname,_,weight) ∈ Part, (scity='Dallas' V scity='Houston'), status<11}

(b)

{(sname) | (snum, _, _) ∈ Supplier, ((snum, pnum, _) ∉ supply V (pnum, _, _, weight) ∉ part V weight <= 10)}

(c)

{(sname, status) | (snum,sname,_,status) ∈ Supplier, (snum, pnum1, _) ∈ Supply, (pnum1, _, 'Green',_) ∈ Part, (snum, pnum2, _) ∈ Supply, (pnum2, _,_,weight) ∈ Part, weight>=10}

(7)

(a)

```
SELECT DISTINCT p.pnum, p.pname, p.weight
FROM Part AS p INNER JOIN Supply AS u ON (p.pnum = u.pnum)
    INNER JOIN Supplier AS s ON (u.snum = s.snum)
WHERE (s.SCity = 'Houston' OR s.SCity = 'Dallas')
AND s.status < 11;
```

(b)

```
SELECT DISTINCT s.sname
FROM supplier AS s
WHERE s.snum NOT IN
       (SELECT u.snum
        FROM supply AS u INNER JOIN part AS p ON (u.pnum = p.pnum)
        WHERE p.Weight >= 10);
```

(c) For example,

```
    SELECT DISTINCT s.sname, s.status
    from supplier AS s INNER JOIN supply AS u1 ON (s.snum = u1.snum)
        INNER JOIN part AS p1 ON (u1.pnum = p1.pnum)
        INNER JOIN supply AS u2 ON (s.snum = u2.snum)
        INNER JOIN part AS p2 ON (u2.pnum = p2.pnum)
    WHERE p1.color = 'Green'
    AND p2.weight >= 10;
```