COP 4710 – Database Systems – Fall 2013

Final Exam Review Sample Problems **SOLUTIONS**

Problem #1 – SQL Queries

Use this sample database:

s (s#, name, rank, city, workers)
p (p#, name, color, weight, city)
j (j#, name, workers, city)
spj (s#, p#, j#, qty)

where: in s: rank is a numeric field, and workers is the number of employees of that supplier.
in p: city is the city in which the part is built.
in j: workers is the number of workers on that job.

A. List the names of all suppliers who supply part number P2 to any job.

```
SELECT  DISTINCT  name
FROM   s
WHERE s#  IN (SELECT s#
                  FROM spj
                  WHERE p# = 'P2');
```

-or-

```
SELECT s.name
FROM  s JOIN spj
WHERE spj.p# = 'P2';
```

B. List the supplier names for those suppliers who do not supply part P2.

```
SELECT  DISTINCT  s.name
FROM s
WHERE  s.s#  NOT IN (SELECT  spj.s#
                           FROM spj
                           WHERE  spj.p# = 'P2');
```

C. List the names of those suppliers who supply at least one red part.

```
SELECT DISTINCT s.name
FROM  s
WHERE s.s# IN (SELECT spj.s#
                   FROM spj
                   WHERE spj.p# IN (SELECT p.p#
                                          FROM p
                                          WHERE p.color = 'red') );
```

D. List all supplier number/part number/ job number triples, such that no two of the indicated supplier, part, or job are located in the same city.

```
SELECT s.s#, p.p#, j.j#
FROM s CROSS JOIN p CROSS JOIN j
WHERE s.city <> p.city AND p.city <> j.city AND s.city <> j.city;
```

E. Get the total quantity of part number P1 that is supplied by supplier number S1.

```
SELECT SUM (spj.qty) AS totalP1byS1
FROM spj
WHERE spj.s# = 'S1' AND spj.p# = 'P1';
```

F. List the part numbers for those parts which are supplied by more than one supplier.

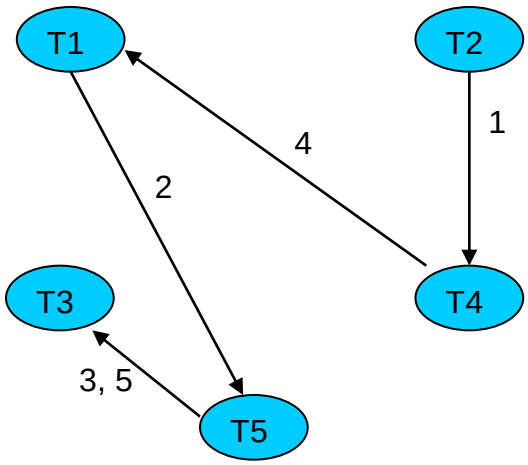
```
SELECT spj.p#
FROM spj
GROUP BY spj.p#
HAVING COUNT (sp.s#) > 1;
```

Problem #2 – Serializability

Shown below is a concurrent schedule S of five transactions operating under an exclusive-locking protocol. Determine if the schedule S is serializable. If the schedule S is serializable, produce a serial schedule equivalent to the concurrent schedule S.

S = [(T1: Xlock A), (T2: Xlock B), (T5: Xlock C), (T2: Unlock B), (T4: Xlock B),
(T1: Unlock A), (T5: Unlock C), (T4: Unlock B), (T5: Xlock A), (T3: Xlock C),
(T1: Xlock B), (T1: Unlock B), (T3: Unlock C), (T5: Unlock A), (T3: Xlock A),
(T3: Unlock A)]

Graph does not contain a cycle – so S is serializable. (T2, T4, T1, T5, T3)



Edge	Reason
1	(T2: Unlock B)…(T4: Xlock B)
2	(T1: Unlock A)…(T5: Xlock A)
3	(T5: Unlock C)…(T3: Xlock C)
4	(T4: Unlock B)…(T1: Xlock B)
5	(T5: Unlock A)…(T3: Xlock A)

Problem #3 – Timestamping protocol

Using the timestamping mechanism for deadlock prevention, we presented two different protocols: "wait or die" and "wound or wait". Given the transaction time stamps ts(T1) = 8, ts(T2) = 4, ts(T3) = 6, and ts(T4) = 2, determine the action for both protocols given the scenarios shown below.

Action	"wait or die" protocol	"wound or wait" protocol
T1 requests an object held by T3	T1: dies T3: continues	T1: waits T3: continues
T2 requests an object held by T1	T2: waits T1: continues	T2: gets lock T1: dies
T4 requests an object held by T2	T4: waits T2: continues	T4: preempts lock T2: dies
T4 requests an object held by T3	T4: waits T3: continues	T4: preempts lock T3: dies
T3 requests an object held by T2	T3: dies T2: continues	T3: waits T2: continues

Problem #4 – Relational Algebra Queries

Construct correct relational algebra expressions for the following queries.

Use this sample database:

s (s#, name, status, city
p (p#, name, color, weight, city)
j (j#, name, workers, city)
spj (s#, p#, j#, qty)

where: in s: status is a numeric field.
in p: city is the city in which the part is built.
in j: workers is the number of workers on that job.

A. List the names of all suppliers who supply part number P2 to any job.

$$\pi_{name}\left(s \triangleright\!\!\triangleleft \left(\pi_{s\#}\left(\sigma_{p\#='P2'}\left(spj\right)\right)\right)\right)$$

B. List the supplier names for those suppliers who do not supply part P2.

$$\pi_{name}(s \triangleright\!\triangleleft ((\pi_{s\#}(\sigma_{p\#\neq'P2'}(spj))) - (\pi_{s\#}(\sigma_{p\#='P2'}(spj)))))$$

C. List the names of those suppliers who supply at least one red part.

$$\pi_{name}(s \triangleright\!\triangleleft (\pi_{s\#}(spj \triangleright\!\triangleleft (\sigma_{color='red'}(p)))))$$

D. List the part names for those parts which are shipped by every supplier.

$$\pi_{name}\left(\left(\pi_{p\#,name}(p)\right) \triangleright\!\triangleleft \left(\left(\pi_{s\#,p\#}(spj)\right) \div \left(\pi_{s\#}(s)\right)\right)\right)$$

E. List all supplier number/part number/ job number triples, such that no two of the indicated supplier, part, or job are located in the same city.

$$\pi_{s\#,p\#,j\#}(\sigma_{s.city\neq p.scity\ AND\ p.city\neq j.city\ AND\ s.city\neq j.city}(s \times p \times j))$$