NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

EXAMINATION FOR
Semester 2 AY2011/12

CS5226 - Database Tuning

April 2012 Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper contains **SEVEN** (7) questions and comprises **SEVEN** (7) printed pages, including this page.
2. Answer **ALL** questions.
3. This is an **OPEN BOOK** examination.
4. Please write your Matriculation Number below.

MATRICULATION NUMBER: _____

EXAMINER'S USE ONLY

Question	Points	Score
1	4	
2	6	
3	7	
4	4	
5	7	
6	8	
7	4	
Total:	40	

1. (4 points) Consider the four relations R, S, and T (where the key attribute of each relation is underlined) and queries Q_1 and Q_2.

   - R(A,B,C)
   - S(D,E,F)
   - T(G,H,I)

Q_1:
```
select  distinct C, D
from    R, S, T
where   E = I
and     C = G
and     B = 10
```

Q_2:
```
select  distinct A, G
from    R, S, T
where   B = H
and     D = 50
and     I = 100
```

   (a) State whether the following statement is *true* or *false*:
   Based on the algorithm discussed in class, we can't conclude that the distinct clause in Q_1 is redundant. ( )

   (b) State whether the following statement is *true* or *false*:
   Based on the algorithm discussed in class, we can't conclude that the distinct clause in Q_2 is redundant. ( )

2. (6 points) Consider the following query Q on the table $R(A, B, C, D, E)$.

```
select  A, C, E
from    R
where   A < 100
and     B = 5
and     C > 20
```

   Write down all possible optimal indexes for Q.

3. (7 points) Consider a database consisting of three relations R, S, and T (where the key attribute of each relation is underlined) and three queries Q_1, Q_2, and Q_3.

   - R(A,B,C,D,E,F)
   - S(I,J,K,L,M,N)
   - T(U,V,W,X,Y,Z)

Q_1:			Q_2:			Q_3:		
	select	C		**select**	I, X		**select**	D, M
	from	R		**from**	S, T		**from**	R, S
	where	A = 100		**where**	I = W		**where**	A = N
				and	L < 30		**and**	F < 50

Suggest a vertical partitioning for each of the tables such that the three queries can be processed efficiently with both the number of redundant data scans as well as the number of joins among partitioned sub-tables minimized.

Your answer should be presented in the following form. For each table, show the schema of each of its partitioned sub-tables. For example, if you are partitioning table R into three sub-tables R_1, R_2, and R_3, then write down their schema: $R_1(\cdots)$, $R_2(\cdots)$, $R_3(\cdots)$.

4. (4 points) Let A be an application that consists of three transactional programs $\{P_1, P_2, P_3\}$ such that $SDG(A)$ does not contain any dangerous structure.

   Suppose we create a new application A' from A by adding an additional program P_4; i.e., A' consists of four transactional programs $\{P_1, P_2, P_3, P_4\}$.

   In the following, note that parts (a) and (c) are not related and they are to be answered independently.

   (a) If all the programs in A contain no update statements (i.e, they are all read-only programs), state whether the following statement is *true* or *false*:
   $SDG(A')$ does not contain any dangerous structure. ( )

   (b) Justify your answer in (a)

   (c) If P_4 does not contain any update statement (i.e., it is a read-only program), state whether the following statement is *true* or *false*:
   $SDG(A')$ does not contain any dangerous structure. ( )

   (d) Justify your answer in (c)

5. (7 points) Consider a database consisting of three relations $R_1(\underline{K_1}, A_1)$, $R_2(\underline{K_2}, A_2)$, and $R_3(\underline{K_3}, A_3)$, where the key of each R_i is K_i. Consider an application that consists of the following four transactional programs, W, X, Y, and Z.

Program W(V)

```
SELECT A1 into :A
FROM R1
WHERE K1 = :V;

UPDATE R2
SET A2 = A2 + 1
WHERE K2 = :A;

COMMIT;
```

Program X(V)

```
UPDATE R2
SET A2 = A2 - 1
WHERE K2 = :V;

COMMIT;
```

Program Y(V)

```
SELECT A2 into :A
FROM R2
WHERE K2 = :V;

UPDATE R3
SET A3 = A3 - 2
WHERE K3 = :A;

COMMIT;
```

Program Z(V)

```
SELECT A2 into :A
FROM R2
WHERE K2 = :V;

SELECT A3 into :B
FROM R3
WHERE K3 = :V;

UPDATE R1
SET A1 = A1 + :B
WHERE K1 = :A;

COMMIT;
```

(a) Show the static dependency graph for this application.

(b) State the minimum number of programs that need to be modified so that the application can become serializable under Snapshot Isolation. Support your answer by showing the necessary modifications for each program to be modified. Note that you only need to show the program statements (i.e., SELECT/UPDATE statements) that are modified; do not repeat any unmodified program statements.

6. (8 points) Consider the following set of transactions $S = \{T_1, T_2, T_3, T_4\}$:

T_1: $R_1(a)$, $R_1(d)$, $W_1(a)$, $R_1(b)$, $W_1(c)$
T_2: $R_2(d)$, $W_2(d)$, $R_2(e)$, $W_2(e)$
T_3: $W_3(f)$, $R_3(g)$, $W_3(g)$
T_4: $W_4(a)$, $R_4(e)$, $R_4(g)$, $W_4(e)$

Write down each of the following:

(a) FineChop(T_1)

(b) FineChop(T_2)

(c) FineChop(T_3)

(d) FineChop(T_4)

7. (4 points) Let chop(T_1), chop(T_2), and chop(T_3) denote a rollback-safe chopping of transactions T_1, T_2, and T_3, respectively.

   Let G_{12} denote the chopping graph consisting of $\{chop(T_1), chop(T_2)\}$, G_{13} denote the chopping graph consisting of $\{chop(T_1), chop(T_3)\}$, G_{23} denote the chopping graph consisting of $\{chop(T_2), chop(T_3)\}$, and G_{123} denote the chopping graph consisting of $\{chop(T_1), chop(T_2), chop(T_3)\}$.

   (i) Which of the following statements is the most appropriate about G_{12}, G_{13}, G_{23}, and G_{123}?

   (a) If each of G_{12}, G_{13}, and G_{23} does not contain any SC-cycle, then G_{123} does not contain any SC-cycle.

   (b) If G_{123} does not contain any SC-cycle, then each of G_{12}, G_{13}, and G_{23} does not contain any SC-cycle.

   (c) (a) or (b)

   (d) (a) and (b)

   (e) None of the above. ( )

   (ii) Justify your answer in (i)

END OF PAPER