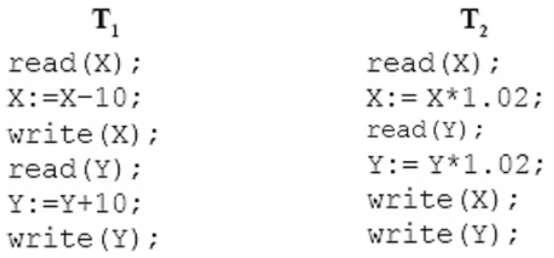
CMPT606 – Advanced Database
Fall 2019 – Homework 3

Due Midnight Thursday 05/12/2019 – Submit your softcopy to blackboard.

Exercise 1 [10 points]

Consider the following 2 transactions T1 and T2:



```
    T1                      T2
read(X);                read(X);
X:=X-10;                X:= X*1.02;
write(X);               read(Y);
read(Y);                Y:= Y*1.02;
Y:=Y+10;                write(X);
write(Y);               write(Y);
```

a. [4 pts] Give a *serial schedule* for these transactions

T_1	T_2
read(X);	
X:=X-10;	
write(X);	
read(Y);	
Y:=Y+10;	
write(Y);	
	read(X);
	X:=X*1.02;
	read(Y);
	Y:=Y*1.02;
	write(X);
	write(Y);

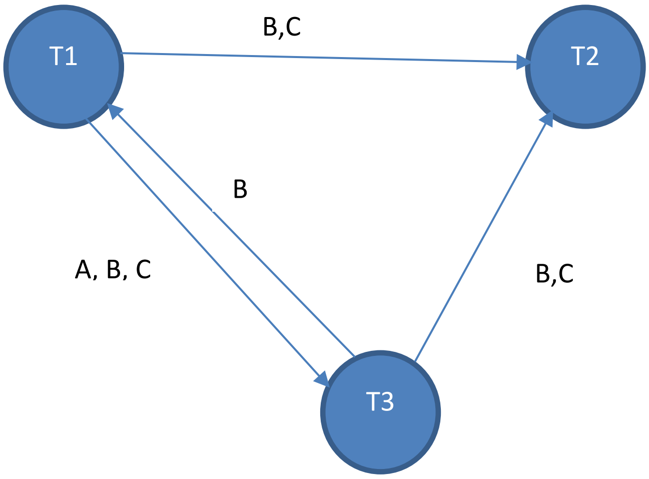
b. [6 pts] Give an *equivalent conflict-serializable schedule* for these transactions.

T1	T2
read(X)	
X:=X-10;	
write(X)	
	read(X);
	X:=X*1.02
read(Y);	
Y:=Y+10	
write(Y)	
	read(Y);
	Y:=Y*1.02;
	write(X);
	write(Y);

Exercise 2 [10 points]
Draw the precedence graph for the following schedule and test whether it is conflict serializable or not.

T1	T2	T3
W(A)		
R(B)		
R(C)		
		W(B)
W(B)		
		W(C)
	R(C)	
	W(B)	
	W(C)	
		R(A)

Solution:



The schedule not conflict serializable because there is a cycle between T1 and T3

Exercise 3 [30 points]

Consider the following transaction log from the start of the execution of a database system that is capable of running undo/redo logging with checkpointing:
(1) < START T1 >
(2) <T1, A, 55, 20>
(3) <T1, B, 255, 20>
(4) <START T2>
(5) <T1, A, 89, 45>
(6) <T2, C, 40, 20>
(7) <COMMIT T1>
(8) <START T3>
(9) <T3, E, 50, 20>
(10) <T2, D, 50, 20>
(11) <START CKPT (T2,T3)>
(12) <T2, C, 70, 30>
(13) <COMMIT T2>
(14) <START T4>

(15) <T4, F, 105, 20>
(16) <COMMIT T3>
(17) <END CKPT>
(18) <T4, F, 155, 95>
(19) <COMMIT T4>

Suppose the log entries are in the format < T_{id}, Variable, Newvalue, Oldvalue >. What is the value of the data items A, B, C, D, E, and F on disk after recovery:

(1) if the system crashes just before line 10 is written to disk?
(2) if the system crashes just before line 13 is written to disk?
(3) if the system crashes just before line 14 is written to disk?
(4) if the system crashes just before line 19 is written to disk?
(5) if the system crashes just after line 19 is written to disk?

Solution:

(1) (A,B,C,D,E,F)=(89,255,20,20,20,20). When the system crashes just before line 10 is written to disk, T1 is committed, and T2 and T3 are incomplete. Thus, we undo T2 and T3 and redo T1.

(2) (A,B,C,D,E,F)=(89,255,20,20,20,20). When the system crashes just before line 13 is written to disk, T1 is committed, and T2 and T3 are incomplete. Thus, we undo T2 and T3 and redo T1.

(3) (A,B,C,D,E,F)=(89,255,70,50,20,20). When the system crashes just before line 14 is written to disk, T1 and T2 are committed, and T3 is incomplete. Thus, we undo T3 and redo T1 and T2.

(4) (A,B,C,D,E,F)=(89,255,70,50,50,20). When the system crashes just before line 19 is written to disk, T1 is committed before START CKPT, thus changes made by T1 must have appear on disk, we can ignore T1. T2 and T3 are committed, and T4 is incomplete, thus we undo T4 and redo T2 and T3 (only for actions after START CKPT, since changes before START CKPT must appear on disk).

(5) (A,B,C,D,E,F)=(89,255,70,50,50,155). When the system crashes just after line 19 is written to disk, T1 can be ignored, and T2, T3, T4 are committed, we redo them (only for actions after START CKPT, since changes before START CKPT must appear on disk).