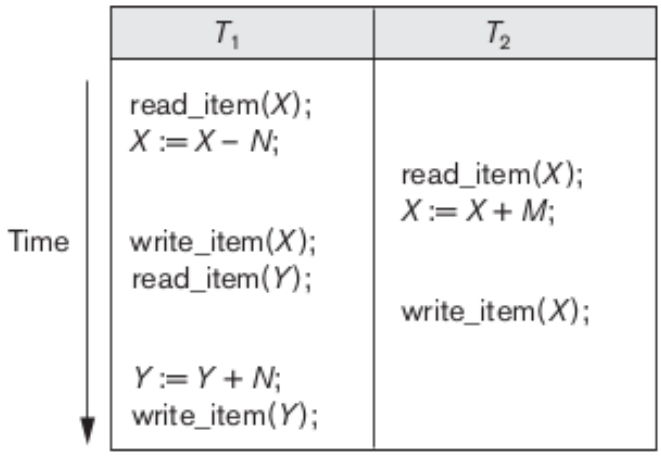


$$S_a: r_1(X);\ r_2(X);\ w_1(X);\ r_1(Y);\ w_2(X);\ w_1(Y);$$

$$S_a: r_1(X);\ r_2(X);\ w_1(X);\ r_1(Y);\ w_2(X);\ c_2;\ w_1(Y);\ c_1;$$

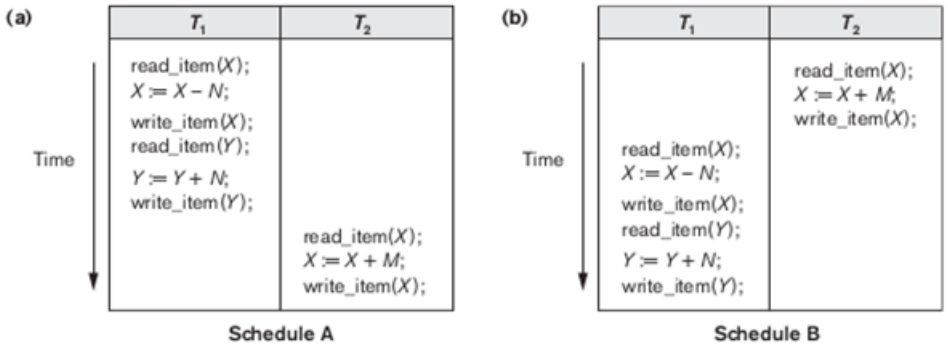
$$S_c: r_1(X);\ w_1(X);\ r_2(X);\ r_1(Y);\ w_2(X);\ c_2;\ a_1;$$

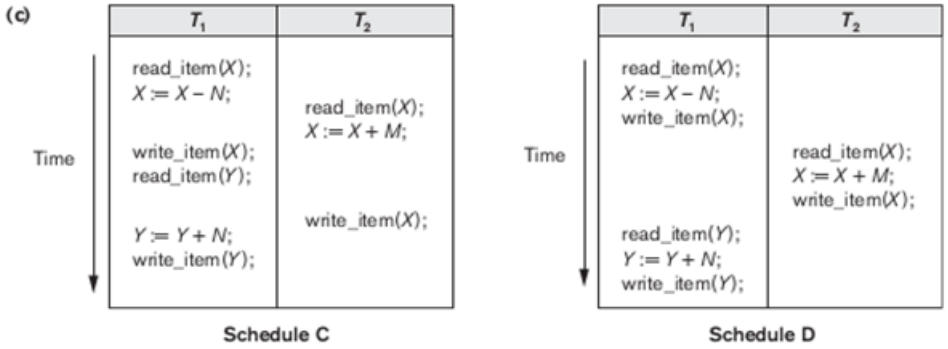
$$S_d: r_1(X);\ w_1(X);\ r_2(X);\ r_1(Y);\ w_2(X);\ w_1(Y);\ c_1;\ c_2;$$

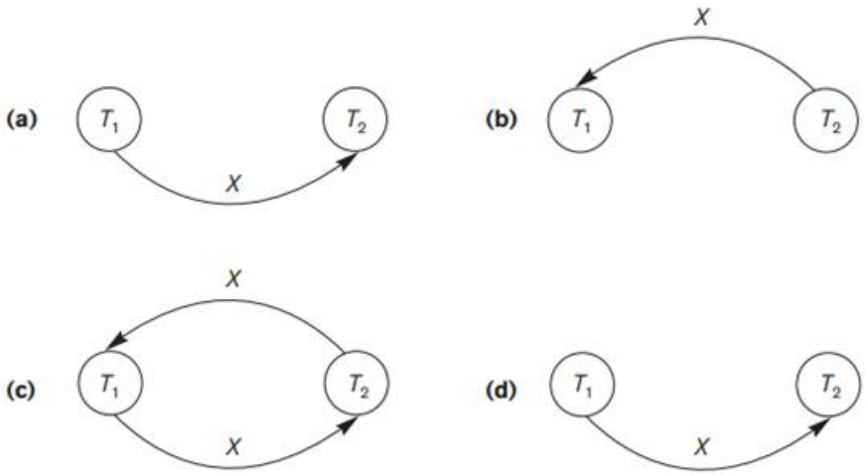
$$S_e: r_1(X);\ w_1(X);\ r_2(X);\ r_1(Y);\ w_2(X);\ w_1(Y);\ a_1;\ a_2;$$

Sf: w1(X, 5); w2(X, 8); a1;

Examples of serial and nonserial schedules involving transactions T_1 and T_2. (a) Serial schedule A: T_1 followed by T_2. (b) Serial schedule B: T_2 followed by T_1. (c) Two nonserial schedules C and D with interleaving of operations.







Constructing the precedence graphs for schedules A to D from to test for conflict serializability. (a) Precedence graph for serial schedule A. (b) Precedence graph for serial schedule B. (c) Precedence graph for schedule C (not serializable). (d) Precedence graph for schedule D (serializable, equivalent to schedule A).

Two schedules that are result equivalent for the initial value of $X = 100$ but are not result equivalent in general.

S_1

read_item(X);
$X := X + 10$;
write_item(X);

S_2
read_item(X); $X := X * 1.1$; write_item (X);

Another example of serializability testing. (a) The read and write operations of three transactions T_1, T_2, and T_3. (b) Schedule E. (c) Schedule F.

(a)

Transaction T_1
read_item(X);
write_item(X);
read_item(Y);
write_item(Y);

Transaction T_2
read_item(Z);
read_item(Y);
write_item(Y);
read_item(X);
write_item(X);

Transaction T_3
read_item(Y);
read_item(Z);
write_item(Y);
write_item(Z);

(b)

Time ↓

Transaction T_1	Transaction T_2	Transaction T_3
	read_item(Z);	
	read_item(Y);	
	write_item(Y);	
		read_item(Y);
read_item(X);		read_item(Z);
write_item(X);		
		write_item(Y);
		write_item(Z);
	read_item(X);	
read_item(Y);		
write_item(Y);	write_item(X);	

Schedule E

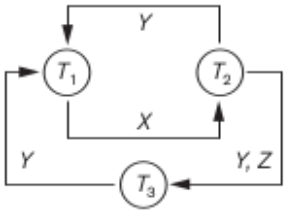
(c)

Time ↓

Transaction T_1	Transaction T_2	Transaction T_3
		read_item(Y);
		read_item(Z);
read_item(X);		
write_item(X);		
		write_item(Y);
		write_item(Z);
	read_item(Z);	
read_item(Y);		
write_item(Y);	read_item(Y);	
	write_item(Y);	
	read_item(X);	
	write_item(X);	

Schedule F

(d)



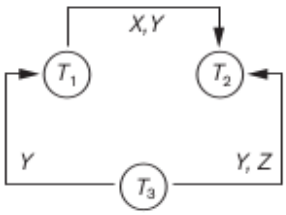
Equivalent serial schedules

None

Reason

Cycle $X(T_1 \rightarrow T_2), Y(T_2 \rightarrow T_1)$
Cycle $X(T_1 \rightarrow T_2), YZ\,(T_2 \rightarrow T_3), Y(T_3 \rightarrow T_1)$

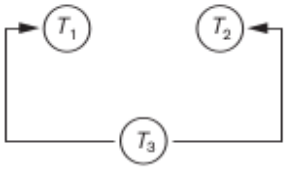
(e)



Equivalent serial schedules

$T_3 \rightarrow T_1 \rightarrow T_2$

(f)



Equivalent serial schedules

$T_3 \rightarrow T_1 \rightarrow T_2$

$T_3 \rightarrow T_2 \rightarrow T_1$

Another example of serializability testing.
(d) Precedence graph for schedule E.
(e) Precedence graph for schedule F.
(f) Precedence graph with two equivalent serial schedules.