

# Rendu TD SGBD

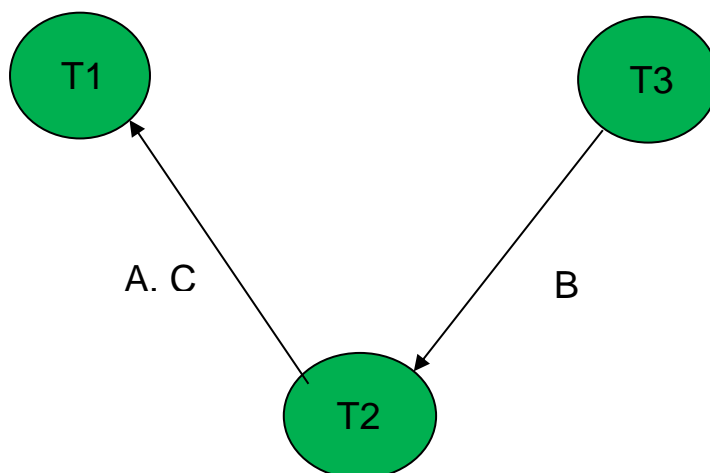
Exercice 4:

- I:

a)  $r1(A)$   $r2(A)$   $r3(B)$   $w1(A)$   $r2(C)$   $r2(B)$   $w2(B)$   $w1(C)$

Soit l'exécution en série :

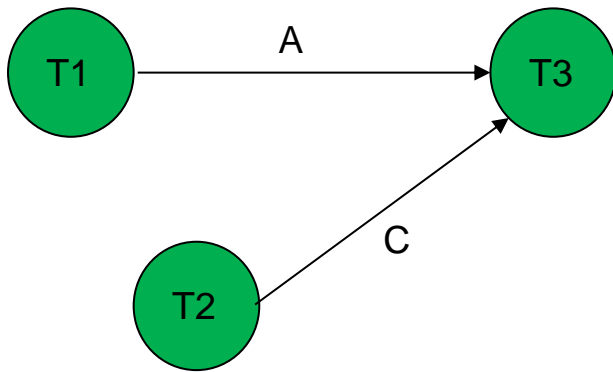
$r3(B)$   $r2(A)$   $r2(C)$   $r2(B)$   $w2(B)$   $r1(A)$   $w1(A)$   $w1(C)$



b)  $r1(A)$   $w1(A)$   $r2(B)$   $w2(C)$   $r3(C)$   $w3(A)$

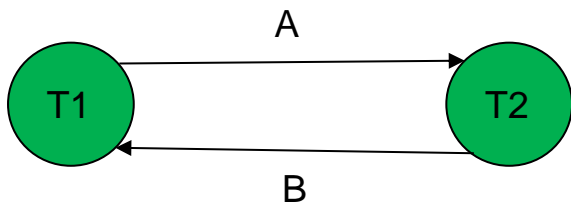
Soit l'exécution en série :

$r1(A)$   $w1(A)$   $r2(B)$   $w2(C)$   $r3(C)$   $w3(A)$   
 $r2(B)$   $w2(C)$   $r1(A)$   $w1(A)$   $r3(C)$   $w3(A)$



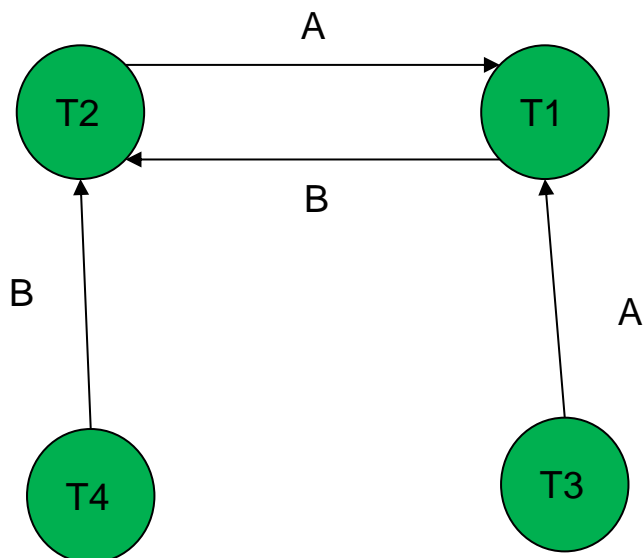
c)  $r1(A)$   $r2(A)$   $w1(B)$   $w2(B)$   $r1(B)$   $r2(B)$   $w2(C)$   $w1(D)$

Il y a un cycle (T1,T2) dans le graphe de dépendance donc pas sérialisable.



d)  $r1(A)$   $r2(A)$   $r1(B)$   $r2(B)$   $r3(A)$   $r4(B)$   $w1(A)$   $w2(B)$

Il y a un cycle (T1,T2) dans le graphe de dépendance donc pas sérialisable.



- II:

Soit E1 et E2 deux exécutions conflict-équivalentes. Si les deux exécutions ont le même nombre de transactions, les deux graphes auront le même nombre de nœud (pareil pour les opérations conflictuelles). L'ordre relatif des opérations conflictuelles reste le même. On en conclut que si les deux exécutions sont conflict-équivalents alors leurs graphes de précedence seront les mêmes.

TD2
-----

### Exercice 1:

- P1 :
  1. Bien formé : avant de faire une opération de lecture ou d'écriture, il faut un lock1 sur la variable 1.  
T1 est bien formée.  
T2 est bien formée.  
T3 est bien formée.
  2. Illégale car l2(B) avant ul1(A)
  3. T1 est 2PL.  
T2 est 2PL.  
T3 est 2PL.
- P2 :
  1. T1 est bien formé.  
T2 est bien formé.  
T3 est bien formé donc P2 est bien formé.
  2. P2 est légal.
  3. P2 est 2PL.

- P3 :
  1. T1 est mal formé car il n'y a pas de verrou sur B, donc P3 est mal formé.
  2. Déverrouillage de B avant de le verrouiller donc T2 est illégal donc P3 est illégal.
  3. Pas de 2PL à cause de  $ul1(B)l2(B)$ .
- P4 :
  1. T1 est bien formée.  
T2 est bien formée.  
T3 est bien formée, donc P4 est bien formée.
  2. P4 est légal.
  3. Nous avons  $ul1(A)$  avant son verrouillage donc P4 n'est pas 2PL.
- P5 :
  1. T1 est bien formée.  
T2 est bien formée.  
T3 est bien formée.
  2. Verrou de T1 sur B alors que T2 essaie de verrouiller donc P5 est illégal.
  3. P5 est 2PL.
- Exercice 2:

**lx** est égal à **lw** et **ls** est égal à **lr** donc **lr** et **lw** sont similaires.

1) r1(A) w1(A) r2(B) w2(B) r1(B) w1(B) r2(A) w2(A)

2) lw2(B) r2(B) w2(B) ul2(B) lw1(A) r1(A) w1(A) ul1(A) lw1(B) r1(B)  
w1(B) ul1(B) lw2(A) r2(A) w2(A) ul2(A)

3) T1: lw1(A) r1(A) w1(A) lw1(B) r1(B) w1(B) ul1(B) ul2(A)  
T2: lw2(B) r2(B) w2(B) lw2(A) r2(A) w2(A) ul2(A) ul1(B)

4) Si une expression est 2PL elle est donc bien formé et légal donc on ne peut pas trouver une exécution non sérialisable.

- Exercice 3:

1) T3 avant aucune des transactions donc T3 ne voit aucune des modifications dans les transactions car il est en mode sérialisable.

2) en fonction de l'isolation de T6, si T6 sérialisable on voit T1 T2 T3 t4 et si non sérialisable T1 T2 T3 T4 T5 T7 T8

3) a) T6 ne peut pas être bloqué par un verrouillage de T3 parce qu'il se situe après la fin de T3 donc tous les verrouillages de T3 ont été déverrouillés.

b) si T8 verrouille une variable avant T6, T6 sera bloqué donc oui il peut être bloqué.