

## Exercice 1 : Les exécutions suivantes sont-elles sérialisables ?

Sérialisable (d'abord les 1, puis les 2...) ?

- Repérer s'il y a des écritures avant les lectures
- Si oui → on peut exécuter les transactions les unes après les autres
- Si non → mécanisme de verrouillage à deux phases
  - Phase d'acquisitions des verrous
  - phase de libération des verrous

Verrou

- En lecture ( $r$ ) → partageable
- En écriture ( $w$ ) → exclusif

Graphe de sériabilité

- Chaque noeud  $\Leftrightarrow$  1 transaction
- Chaque arc  $\Leftrightarrow$  dépendance entre les transactions

Règles :

$1 \rightarrow_x 2$

- $r1(x)r2(x) \Leftrightarrow r2(x)r1(x)$
- $r1(x)w2(x) \neq w2(x)r1(x)$
- $w1(x)r2(x) \neq r2(x)w1(x)$
- $w1(x)w2(x) \neq w2(x)w1(x)$

1.  $r1(x), w1(x), r2(z), w1(t), r1(y), w1(y), r2(x), r2(t), w2(x), w2(z)$

$1 \rightarrow_{x,t} 2$

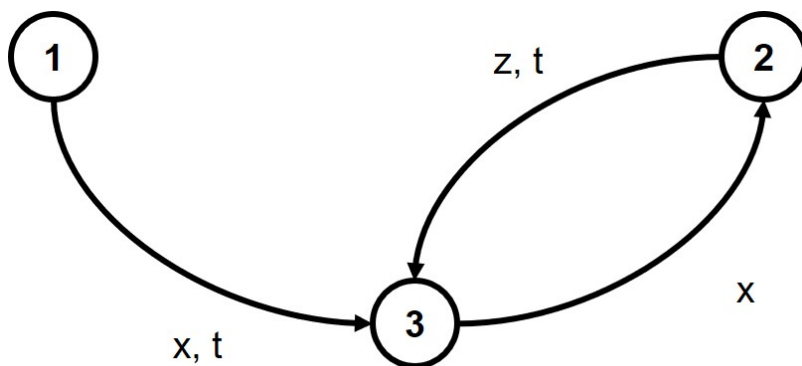
Pas de cycle → exécution en série équivalente.

⇒ sérialisable

$r1(x), w1(x), w1(t), r1(y), w1(y), r2(z), r2(x), r2(t), w2(x), w2(z)$

2.  $w1(x), r2(y), w2(z), w3(x), w2(x), r3(z), r1(t), r2(t), w3(t)$

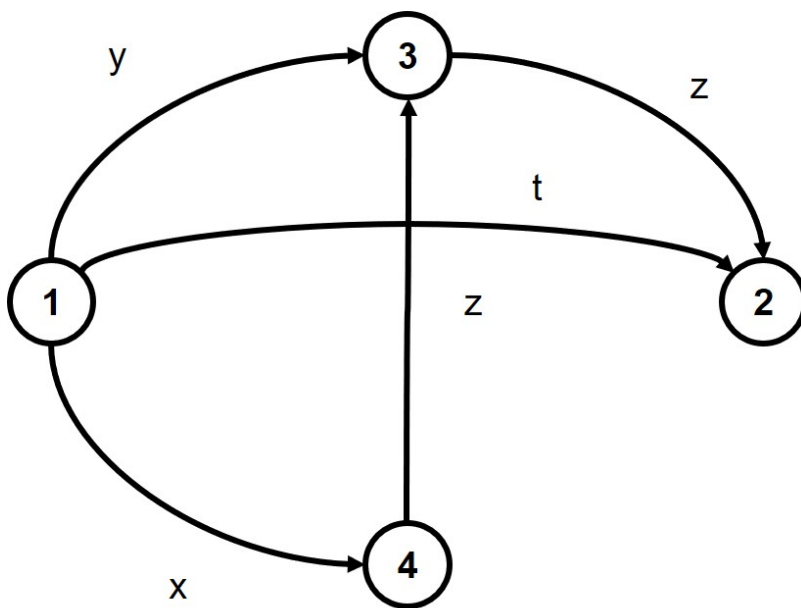
Graphe de sérialisation :



Cycle ⇒ pas sérialisable.

3.  $r1(x), r4(x), w4(x), r1(y), r4(z), w4(z), w3(y), w3(z), w1(t), w2(z), w2(t)$

Graphe de sérialisation :



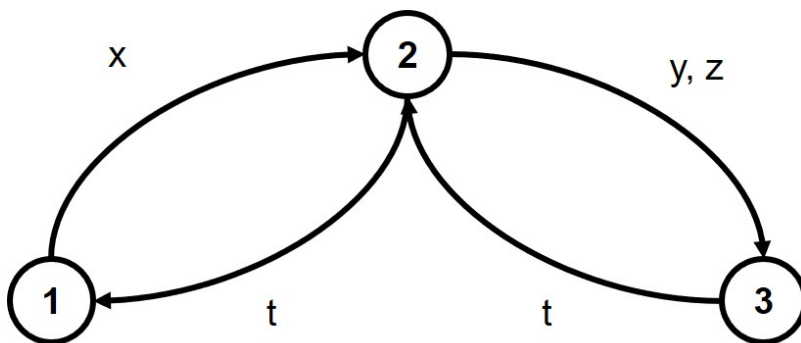
Pas de cycle  $\Rightarrow$  sérialisable.

Pour sérialiser on commence par la 1, puis la 4, puis la 3 et enfin la 2 :

- $r1(x), r1(y), w1(t), r4(x), w4(x), r4(z), w4(z), w3(y), w3(z), w2(z), w2(t)$

4.  $r1(x), r2(y), w1(x), r3(y), r2(x), w3(y), r2(t), r3(t), w2(z), w3(z), w2(t), w1(t)$

Graphe de sérialisation :



Il y a des cycles  $\Rightarrow$  non-sérialisable. On ne peut commencer sur aucun élément, tous voient des départs et des arrivées.

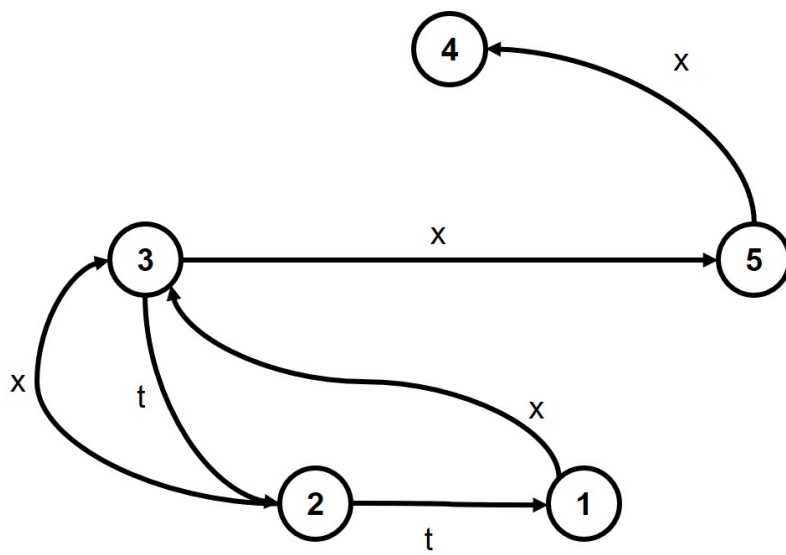
**Exercice 2 :** Les transactions T1, T2, T3, T4 et T5 s'exécutent de manière concurrente. Soit H un historique d'exécution :

$H = T1 : read(X); T2 : read(X); T3 : write(X); T3 : write(T); CommitT3; T4 : write(Y); T5 : write(X); T2 : read(t); T4 : read(X); CommitT5; T2 : write(T); T1 : read(T); CommitT1; T4 : read(Z); T2 : read(Z); CommitT2; CommitT4$

1. L'exécution  $H$  a-t-elle une exécution en série équivalente ?

$\Rightarrow r1(x), r2(x), w3(x), w3(t), c3, w4(y), w5(x), r2(t), r4(x), c5, w2(t), r1(t), c1, r4(z), r2(z), c2, c4$

Graphe de sériabilité



2. Donnez l'exécution obtenue en utilisant le verrouillage à 2 phases.

On évolue en suivant le cycle :

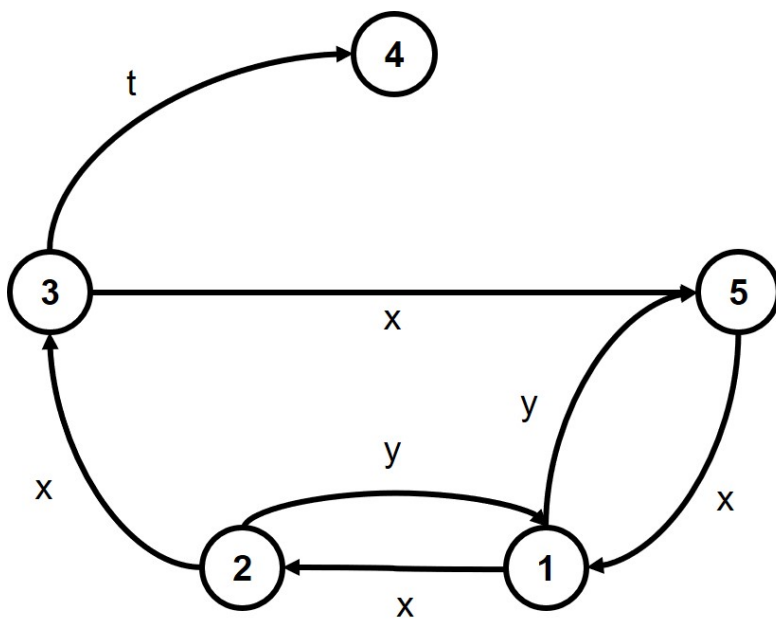
- $r1(x) \rightarrow lect_1(x)$
- $r2(x) \rightarrow lect_{1,2}(x)$
- $w3(x) \rightarrow$  Il ya a déjà un verrou en lecture sur  $x$ . L'entièreté de la transaction  $T3$  est bloquée.
- $\Rightarrow w3(t) \rightarrow T3$  bloquée.
- $\Rightarrow c3 \rightarrow T3$  bloquée.
- $w4(y) \rightarrow ecrit_4(y)$
- $w5(x) \rightarrow T5$  est bloquée.
- $\Rightarrow c5 \rightarrow T5$  bloquée.
- $r2(t) \rightarrow lect_2(t)$
- $w2(t) \rightarrow ecrit_2(t)$  On est sur la même transaction  $\Rightarrow$  on change le verrou lecture en écriture. Si on a un verrou en écriture on peut quand même lire dessus donc on ne transforme pas un verrou en écriture en verrou en lecture
- $r1(t) \rightarrow T1$  bloquée.
- $\Rightarrow c1 \rightarrow T1$  bloquée.
- $r4(z) \rightarrow lect_4(z)$
- $r2(z) \rightarrow lect_{2,4}(z)$
- $c2 \rightarrow$  on relâche les verrous posés par  $T2$
- $c4 \rightarrow$  on relâche les verrous posés par  $T4$
- $w3(x) \rightarrow T3$  est bloquée
- $w3(t) \rightarrow T3$  bloquée.
- $c3 \rightarrow T3$  bloquée.
- $w5(x) \rightarrow T5$  est bloquée.
- $c5 \rightarrow T5$  bloquée.
- $r1(t) \rightarrow lect_1(t)$
- $c1 \rightarrow$  on relâche les verrous posés par  $T1$
- $w3(x) \rightarrow ecrit_3(x)$
- $w3(t) \rightarrow ecrit_3(t)$
- $c3 \rightarrow$  on relâche les verrous posés par  $T3$
- $w5(x) \rightarrow ecrit_5(x)$
- $c5 \rightarrow$  on relâche les verrous posés par  $T5$

17 opérations

**Exercice 2 bis :** Même questions pour  $H'$

$H' = r1(x), w2(x), w3(x), r3(t), c3, w2(y), c2, w1(y), w4(t), c4, w5(y), w5(x), c5, w1(x), c1$

**Graphe de sériabilité**



- $r1(x) \rightarrow lect_1(x)$ , Première opération
- $w2(t) \rightarrow T2$  bloquée.
- $w3(t) \rightarrow T3$  bloquée.
- $r3(t) \rightarrow T3$  bloquée.
- $c3 \rightarrow T3$  bloquée.
- $w2(y) \rightarrow T2$  bloquée.
- $c2 \rightarrow T2$  bloquée.
- $w1(y) \rightarrow ecrit_1(y)$ , Deuxième opération
- $w4(t) \rightarrow ecrit_4(t)$ , Troisième opération
- $c4 \rightarrow$  on relâche les verrous posés par  $T4$ , Quatrième opération...
- $w5(y) \rightarrow T5$  bloquée.
- $w5(x) \rightarrow T5$  bloquée.
- $c5 \rightarrow T5$  bloquée.
- $r1(x) \rightarrow ecrit_1(x)$
- $c1 \rightarrow$  on relâche les verrous posés par  $T1$
- $w2(x) \rightarrow ecrit_2(x)$
- $w3(x) \rightarrow T3$  bloquée.
- $r3(t) \rightarrow T3$  bloquée.
- $c3 \rightarrow T3$  bloquée.
- $w2(y) \rightarrow ecrit_2(y)$
- $c2 \rightarrow$  on relâche les verrous posés par  $T2$
- $w5(y) \rightarrow ecrit_5(y)$
- $w5(x) \rightarrow ecrit_5(x)$
- $c5 \rightarrow$  on relâche les verrous posés par  $T5$
- $w3(x) \rightarrow ecrit_3(x)$
- $r3(t) \rightarrow lect_3(x)$
- $c3 \rightarrow T3$  on relâche les verrous posés par  $T3$

15 opérations