

# Systemes d'exploitations II

## Présentation #8 : L'interblocage

12/12/2021

Ahmed Benmoussa



Centre universitaire d'Aflou

# Interblocage (*Deadlocks*)

**Définition:** *un ensemble de processus sont en inter-blocage si chacun des processus attend un évènement que seul un processus de cet ensemble peut satisfaire.*

L'interblocage peut avoir plusieurs scénarios, le plus simple étant le **blocage mutuel** (*mutually recursive locking*).

# mutually recursive locking

// Thread A

lock1.acquire();

lock2.acquire();

lock2.release();

lock1.release();

**attend**

// Thread B

lock2.acquire();

lock1.acquire();

lock1.release();

lock2.release();



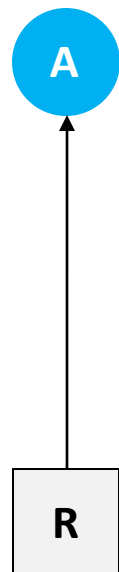
# Les conditions d'interblocage

- Coffman et al. (1971) ont défini 04 conditions pour qu'il est un interblocage:
  1. **Exclusion mutuelle** si une ressource n'est libre, elle doit être assignée à un et un seul processus.
  2. **Détenir et attendre (Hold-and-wait)** chaque processus détient au moins une ressource et attend une autre ressource
  3. **Non-préemption** Les ressources allouées à un processus ne peuvent être lui retirées que s'il la libère.
  4. **Attente circulaire** L'existence d'une liste circulaire de deux processus ou plus attendant une ressource détenu par le prochain processus de cette chaîne

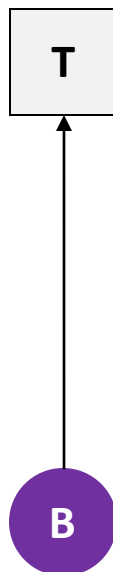
# Représentation d'interblocages

- Holt (1972) a proposé de représenter l'allocation des ressources par des graphes orientés.
- Les processus sont représentés par des cercles.
- Les ressources par des rectangles.

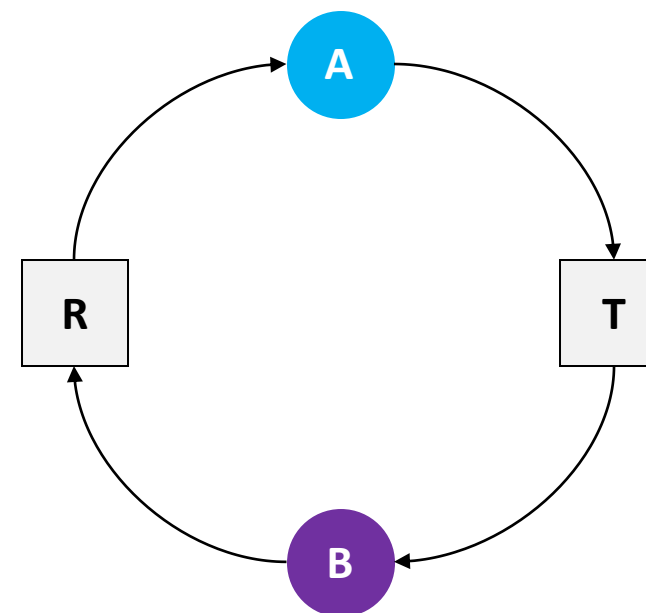
# Graphes d'allocation des ressources



Le **processus A**  
détient la  
ressource **R**



Le **processus B**  
attend la  
ressource **T**



**Interblocage.**  
Le cycle est: **B-R-A-T-B**

# Exemple

- Trois processus: A, B et C
- Trois ressources: R, S et T

A	B	C
Demande R	Demande S	Demande T
Demande S	Demande T	Demande R
Relâche R	Relâche S	Relâche T
Relâche S	Relâche T	Relâche R

# Exemple 1

- Si l'exécution est séquentielle, pas de course à la ressource. Donc pas d'interblocage.
- Supposant l'exécution se passe en cet ordre:

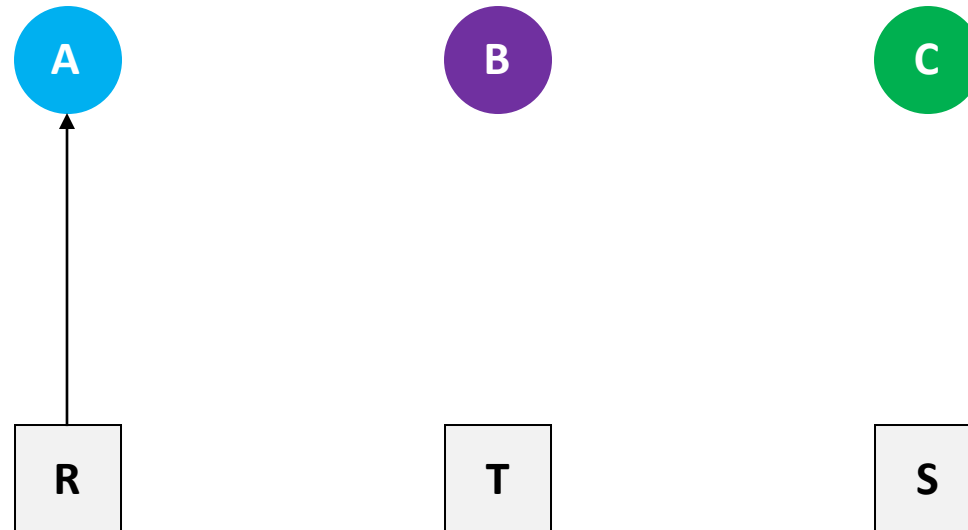
1. A demande R
2. B demande S
3. C demande T
4. A demande S
5. B demande T
6. C demande R

A	B	C
Demande R	Demande S	Demande T
Demande S	Demande T	Demande R
Relâche R	Relâche S	Relâche T
Relâche S	Relâche T	Relâche R



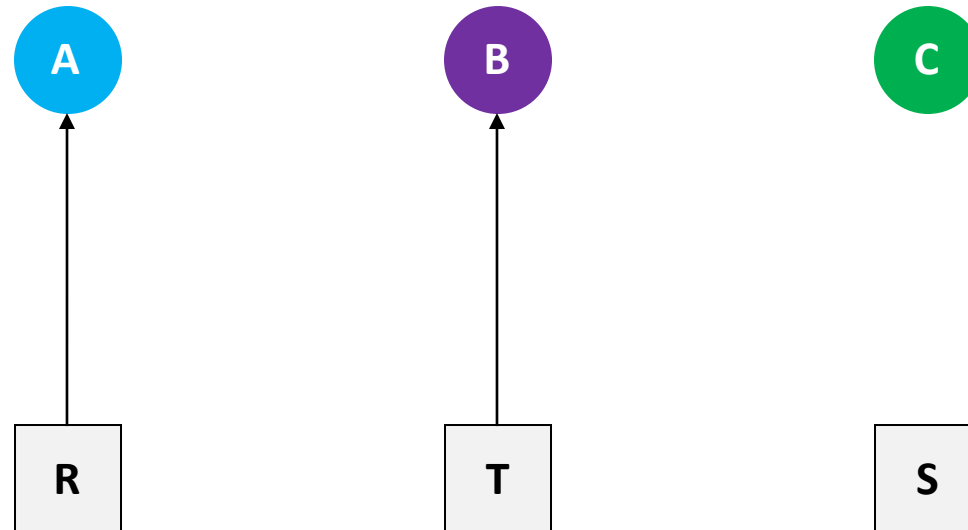
# Exemple

1: A demande R



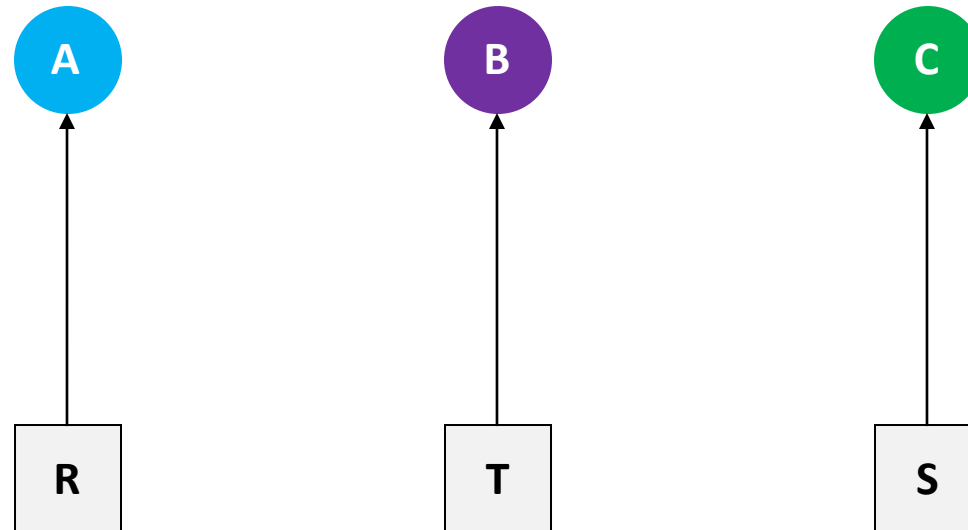
# Exemple

2: B demande T



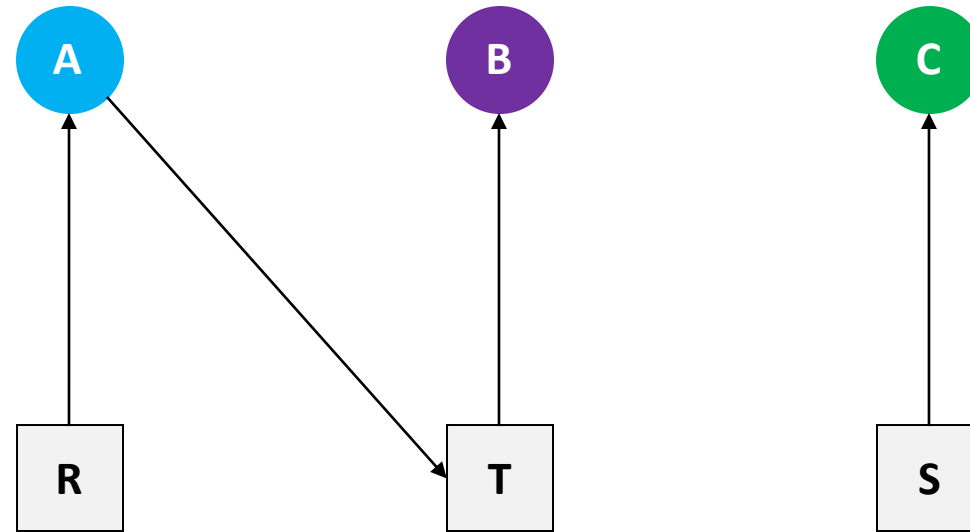
# Exemple

3: C demande S



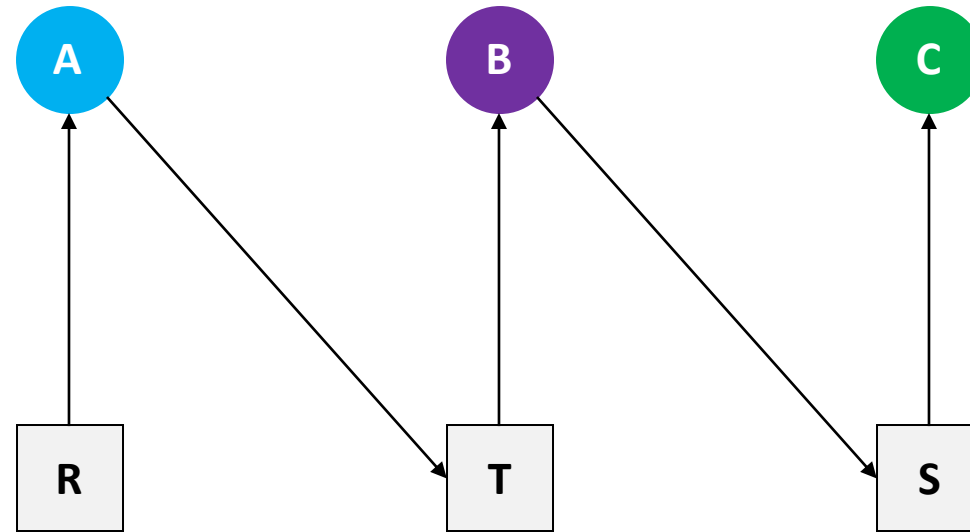
# Exemple

4: A demande T



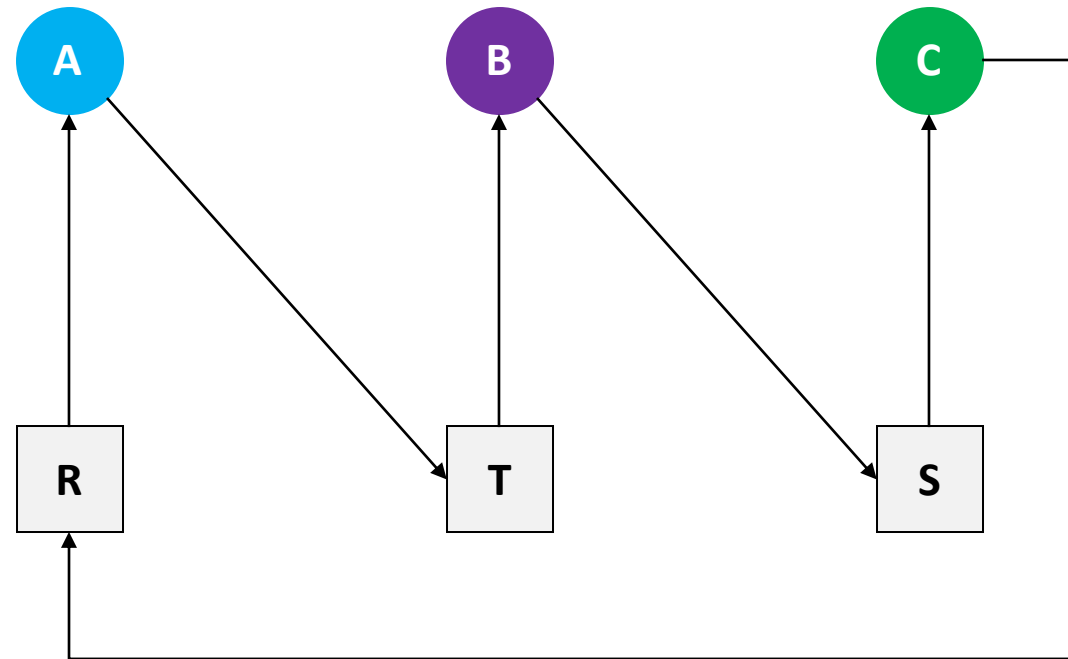
# Exemple

5: B demande S

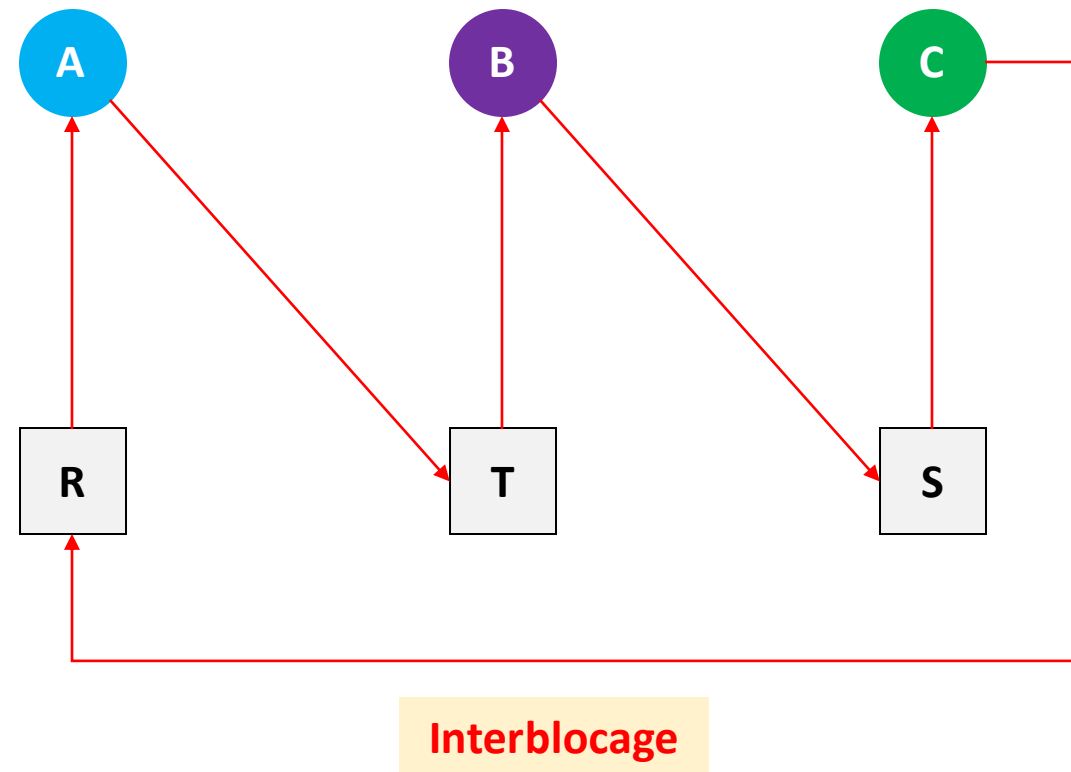


# Exemple

6: C demande R



# Exemple



# Ordonnancement des processus

- Le système d'exploitation n'exécute pas les processus dans un ordre précis.
- Il peut suspendre l'acquisition d'une ressource si elle va conduire à un interblocage.
- Il peut aussi choisir de ne pas exécuter un processus tant qu'il n'est pas fiable.



## Exemple 2

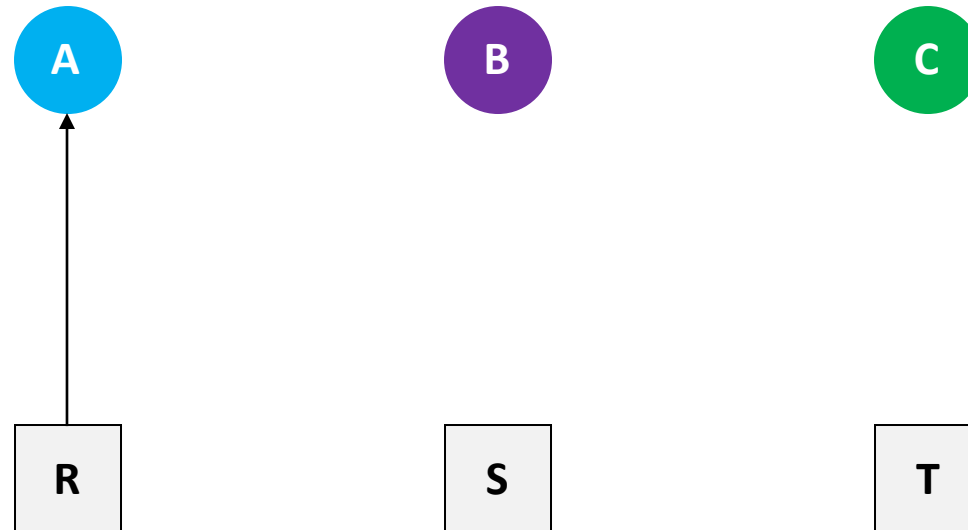
- Le système choisi de retarder l'exécution du processus B
- L'exécution se passe comme suit:

1. A demande R
2. C demande T
3. A demande S
4. C demande R
5. A relache R
6. A relache S

A	B	C
Demande R	Demande S	Demande T
Demande S	Demande T	Demande R
Relâche R	Relâche S	Relâche T
Relâche S	Relâche T	Relâche R

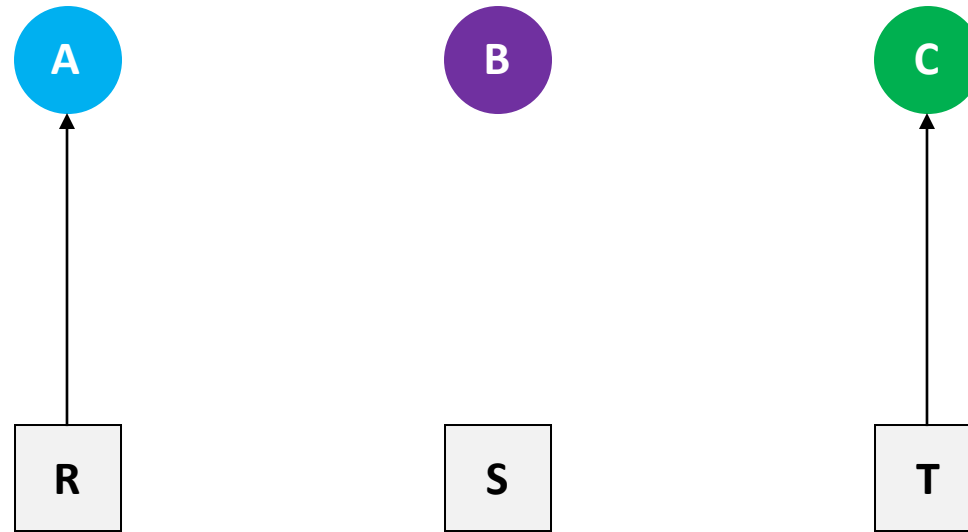
## Exemple N°2

1: A demande R



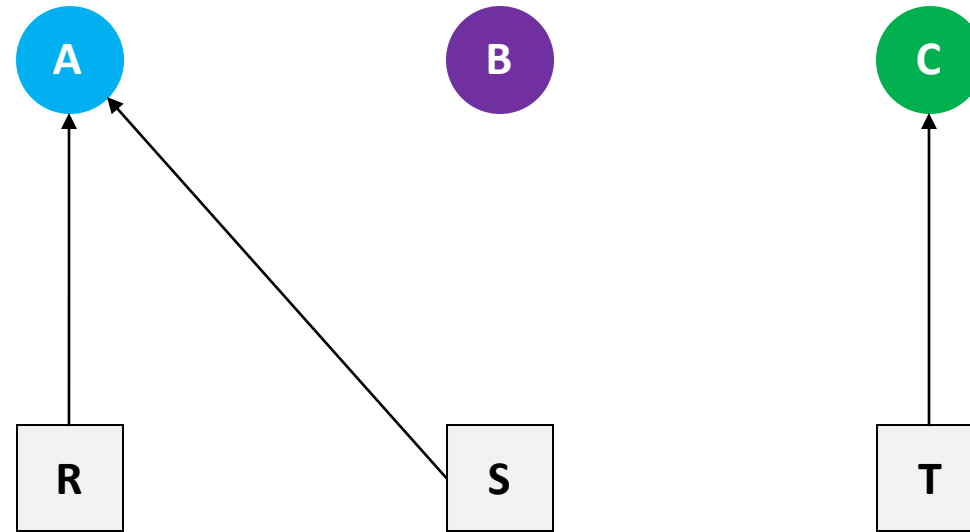
## Exemple N°2

2: C demande T



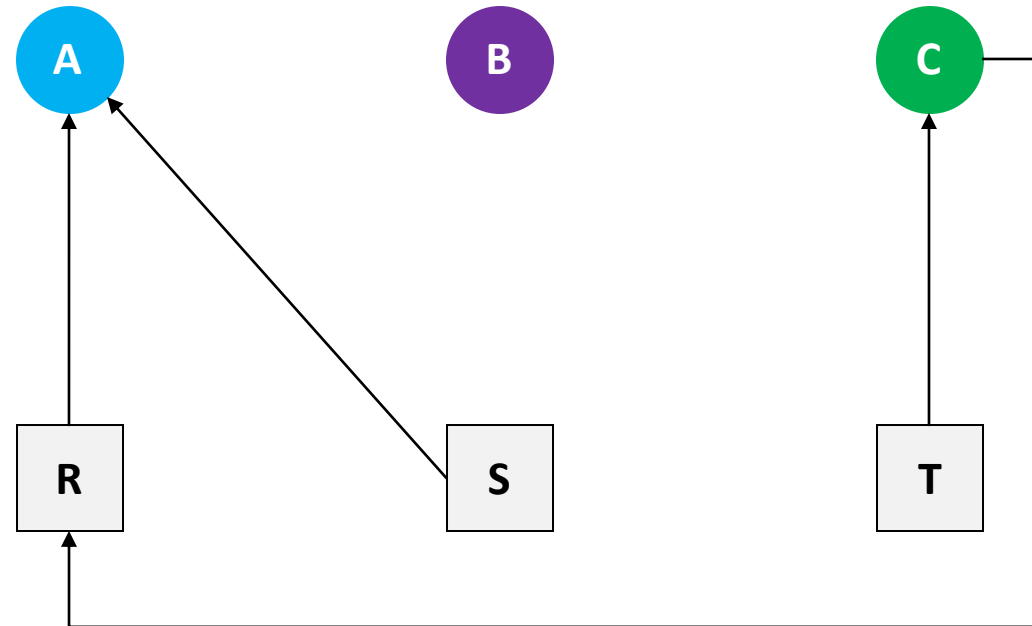
## Exemple N°2

3: A demande S



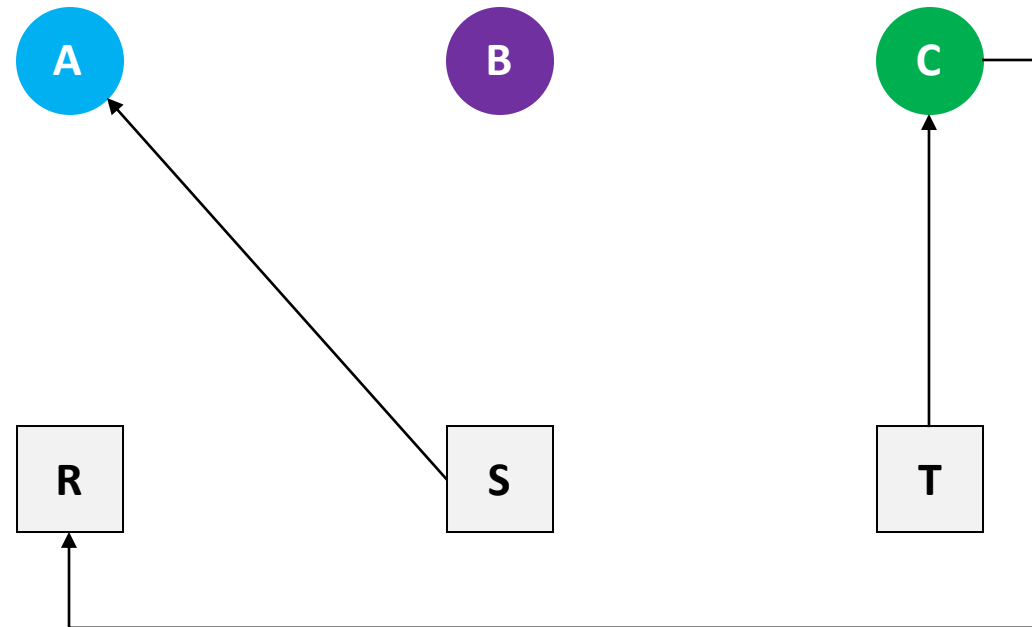
## Exemple N°2

4: C demande R



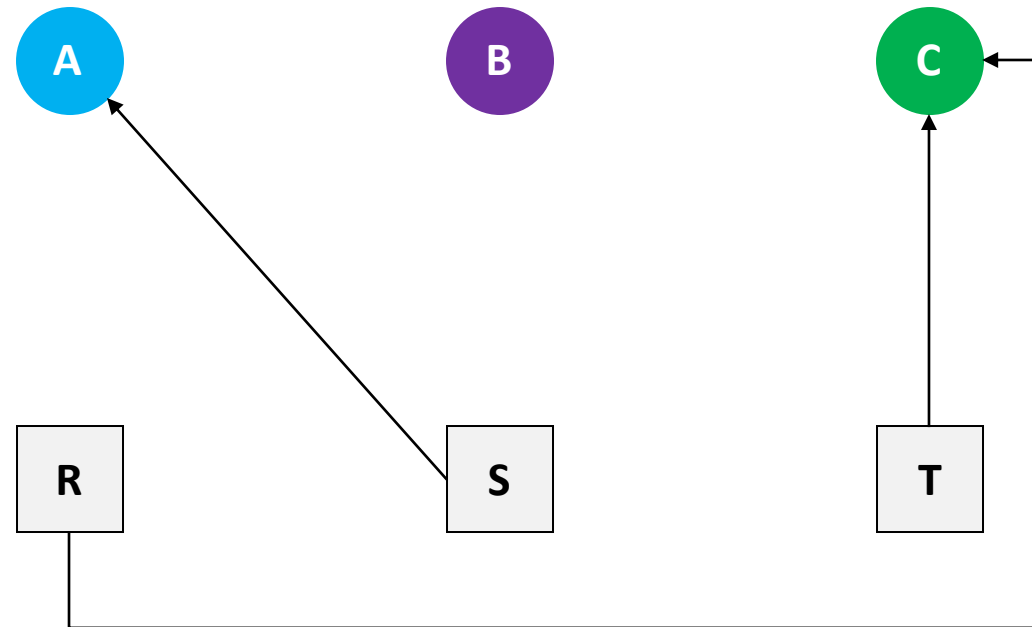
## Exemple N°2

4: A relache R



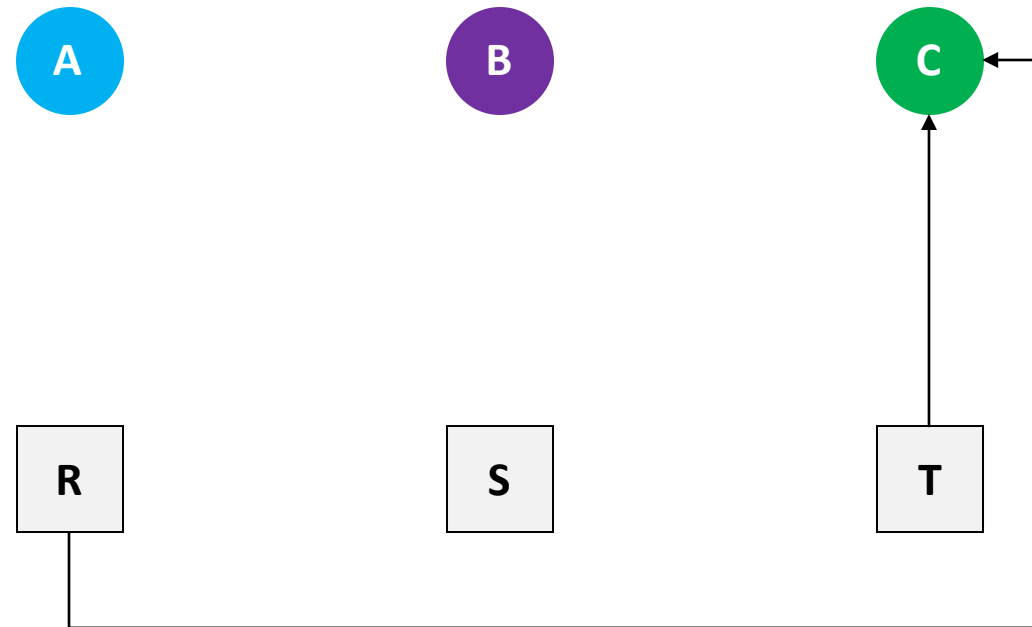
## Exemple N°2

4: C rdetient R



## Exemple N°2

4: A relache S



Pas  
d'interblocage



# Interblocage: Stratégies

1. Ignorer le problème, peut il vous ignorera.
2. **Détection et récupération**: laisser arriver, détecter et perdre actions.
3. **Evitement dynamique**: allouer les ressources avec prudence.
4. **Prévention**: en négligeant l'une des quatres conditions citées plus haut.

# L'algorithme de l'autruche

- Ignorer le problème.
- Deux vision: les mathématiciens et les ingénieurs.

# Interblocage: Stratégies

1. Ignorer le problème, peut il vous ignorera.
2. **Détection et récupération:** laisser arriver, détecter et perdre actions.
3. **Evitement dynamique:** allouer les ressources avec prudence.
4. **Prévention:** en négligeant l'une des conditions citées plus haut.

## II. Détection et récupération

- Le système n'empêche pas l'apparition d'un interblocage.
- Il tente de détecter quand sa arrive.
- Ensuite il prend des actions nécessaires afin de le gérer.
- **Deux type:**
  1. interblocage avec une ressource de chaque type.
  2. Interblocage avec multiples ressources de chaque type.

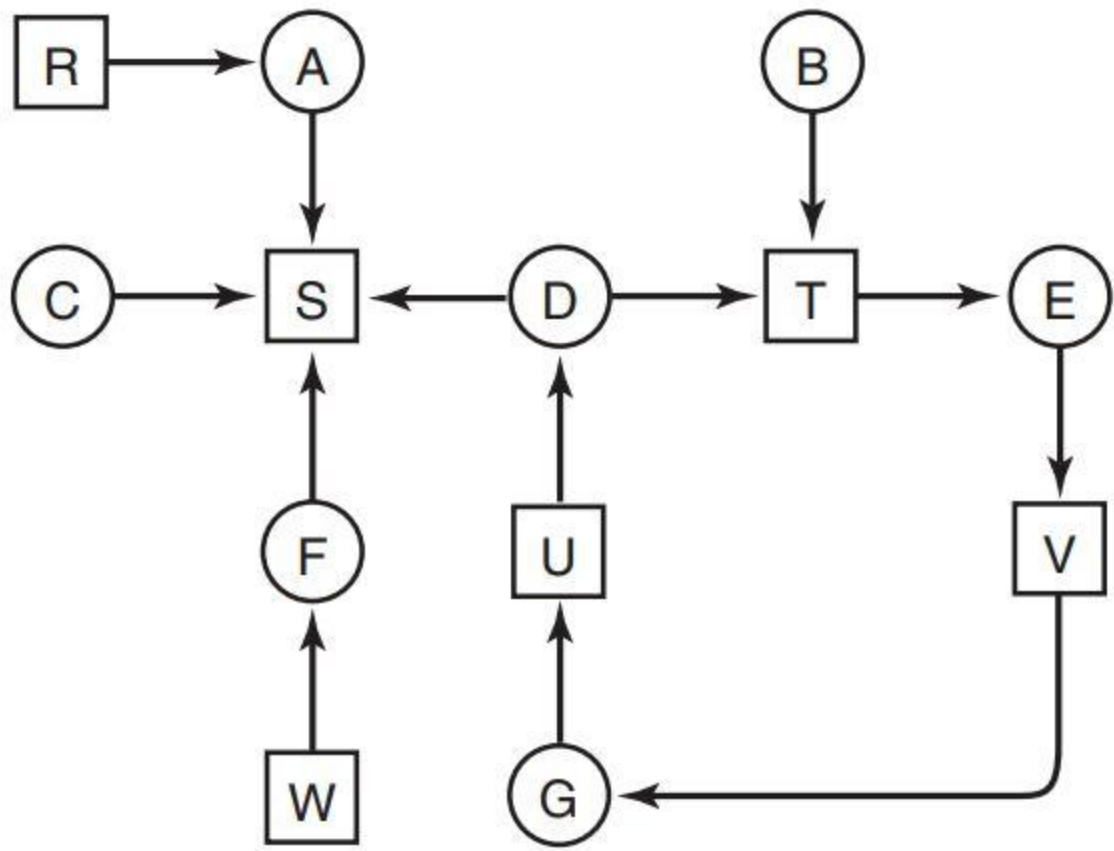
# Interblocage avec une ressource de chaque type

- Une ressource de chaque type: imprimante, lecteur CD, etc...
- Si le graphe d'allocation des ressource contient un cycle (comme l'exemple précédent), cela signifie l'existence d'un interblocage.
- Chaque processus a l'interieur du cycle est **interbloqué**.

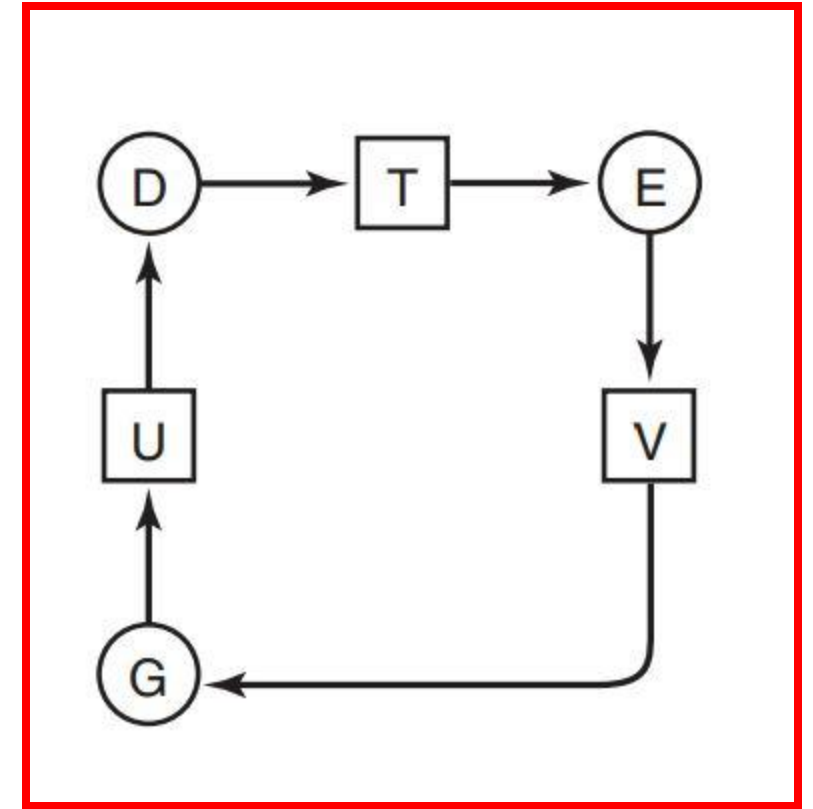
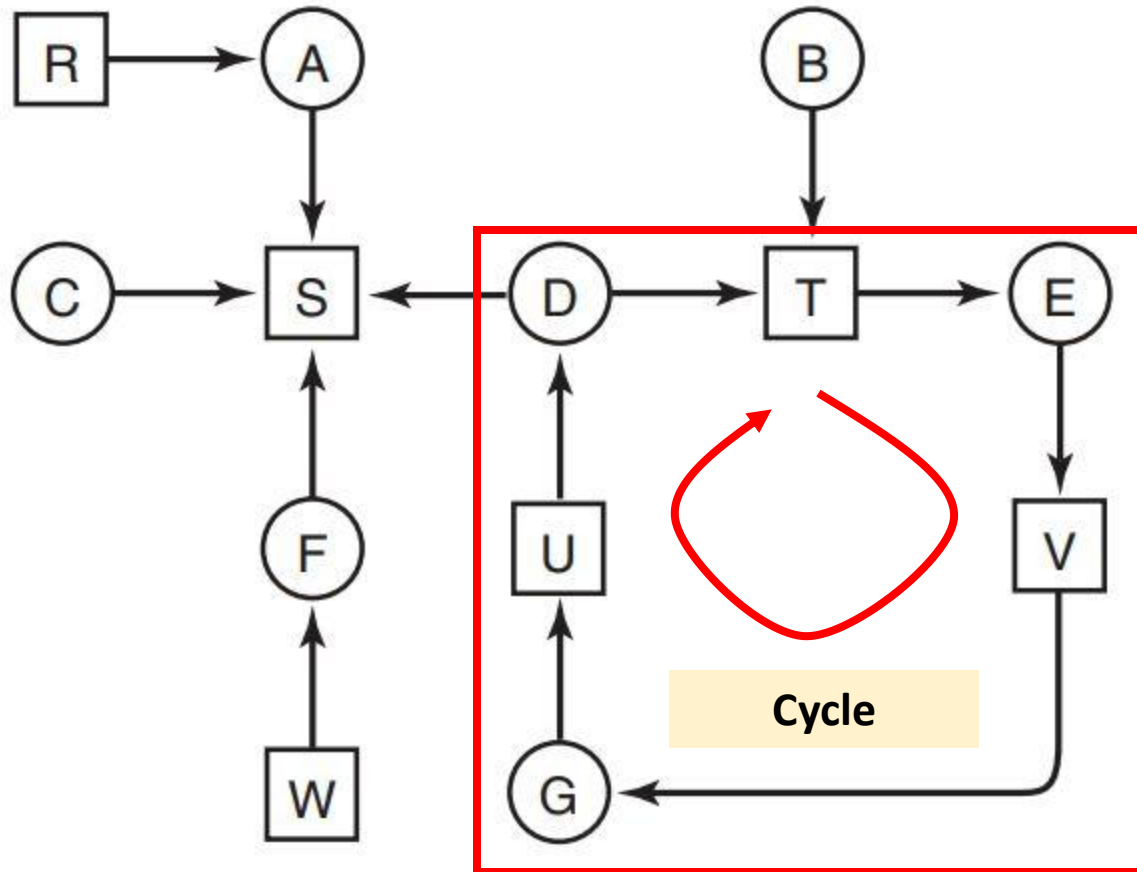
# Exemple

- Supposant **sept (07) processus** de A à G.
- Et six **(06) ressources**.
  1. Processus *A* détient *R* et veut *S*.
  2. Processus *B* veut *T*.
  3. Process *C* veut *S*.
  4. Process *D* détient *U* et veut *S* and *T*.
  5. Process *E* détient *T* and veut *V*.
  6. Process *F* détient *W* and veut *S*.
  7. Process *G* détient *V* and veut *U*.

# Exemple: graphe d'allocation des ressources



## Exemple: graphe d'allocation des ressources





# Guérison d'un interblocage

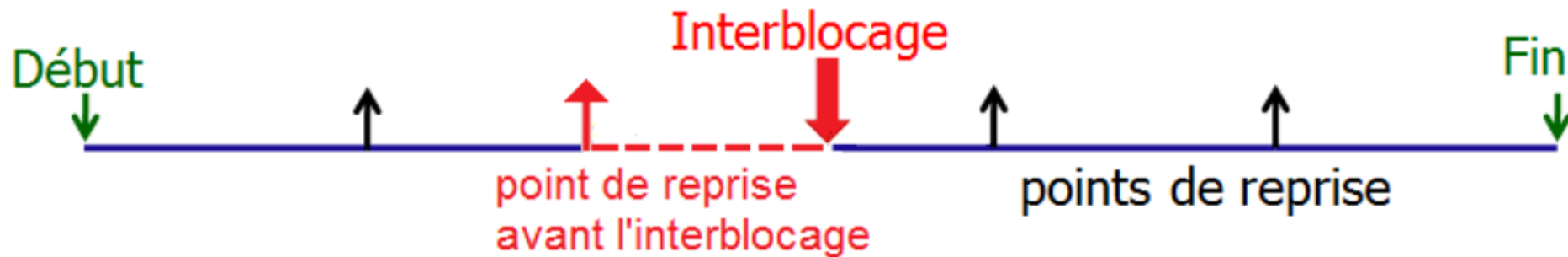
- Supposant on a détecté un interblocage, on doit revenir a un état sain.
- **Guérison avec préemption**
- **Retour en arrière (*Rollback*)**
- **Terminer de processus**

# Guérison avec préemption

- Retirer une ressource a un processus et la donner à un autre processus afin de débloquent.
- Dépend de la ressource a retirer
- Revenir à l'état précédent est difficile voire impossible.
- Choisir le processus a suspendre dépend de la ressource qu'il détient.

# Guérison avec retour en arrière (Rollback)

- Ce mode de guérison repose sur les points de reprise (*checkpoints*)
- Le programme du processus victime est ré-exécuté à partir du dernier point de reprise effectué avant l'interblocage.
- Cette solution est moins couteuse.



# Guérison avec retour en arrière (Rollback)

- Les processus déclarent des points de reprise à des endroits précis (jugés importants) de leur code exécutable.
- A chaque passage par l'un de ces points, le système mémorise (sauvegarder sur disque) une image complète de l'espace mémoire occupé par le processus et de l'état des ressources utilisées par le processus de façon à pouvoir reprendre son exécution dans un état cohérent.
- Cette technique est très utilisée dans les SGBD (journal des transactions).

# Guérison en tuant des processus

- La méthode la plus simple consiste à terminer un processus (tuer le processus).
- Le processus à tuer dépend aussi de la ressource qu'il détient.

# Interblocage: Stratégies

1. Ignorer le problème, peut il vous ignorera.
2. **Détection et récupération:** laisser arriver, détecter et perdre actions.
3. **Evitement dynamique:** allouer les ressources avec prudence.
4. **Prévention:** en négligeant l'une des conditions citées plus haut.

# Algorithme du banquier

- Presenter par *Dijkstra* (1965), est un algorithme d'ordonnancement.
- Vérifie si une requête conduit a un etat sain, elle approuvée sinon elle est retardée.

# Etats sains et non sains

Has Max		
A	3	9
B	2	4
C	2	7

Free: 3

Has Max		
A	3	9
B	4	4
C	2	7

Free: 1

Has Max		
A	3	9
B	0	–
C	2	7

Free: 5

Has Max		
A	3	9
B	0	–
C	7	7

Free: 0

Has Max		
A	3	9
B	0	–
C	0	–

Free: 7

**Etat sain**



## Exemple N°2

Has Max		
A	3	9
B	2	4
C	2	7

Free: 3

Has Max		
A	4	9
B	2	4
C	2	7

Free: 2

Has Max		
A	4	9
B	4	4
C	2	7

Free: 0

Has Max		
A	4	9
B	—	—
C	2	7

Free: 4

**Etat non sain**

**Etat non sain != interblocage**

# Exemple

$$E = \begin{pmatrix} 4 & 2 & 3 & 1 \end{pmatrix}$$

Tape drives  
Plotters  
Scanners  
Blu-rays

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 \end{pmatrix}$$

Tape drives  
Plotters  
Scanners  
Blu-rays

Current allocation matrix

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$

Request matrix

$$R = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix}$$

## Exemple N°2

$$E = \begin{pmatrix} 4 & 2 & 3 & 1 \end{pmatrix}$$

Tape drives  
Plotters  
Scanners  
Blu-rays

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 \end{pmatrix}$$

Tape drives  
Plotters  
Scanners  
Blu-rays

Current allocation matrix

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$

Request matrix

$$R = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{bmatrix}$$

Interblocage