

**Exercice 1 :**

On suppose que l'exécution d'un processus se compose de deux temps processeur entre les quels s'intercale une demande d'E/S. On suppose de plus que les processus n'attendent pas pour leur E/S (par exemple, ils utilisent tous un périphérique différent).

Les processus usuellement utilisés par le système après son chargement sont :

Processus	Arrivé	Temps CPU #1	Temps E/S	Temps CPU #2	Priorité
A	0	9 ms	2 ms	3 ms	3
B	4	6 ms	5 ms	3 ms	2
C	6	4 ms	4 ms	7 ms	1
D	8	2 ms	6 ms	8 ms	2
E	10	4 ms	3 ms	3 ms	1
F	12	5 ms	1 ms	2 ms	3

**Question 1 :** Donnez l'assignation pour l'ordonnancement des ces processus pour une stratégie FIFO et calculer le temps moyen de séjour (temps de rotation).

**Question 2 :** Donnez l'assignation pour l'ordonnancement des ces processus pour une stratégie avec Priorité (3 étant le plus niveau) et calculer le temps moyen de séjour (temps de rotation).

**Question 3 :** Donnez l'assignation pour l'ordonnancement des ces processus pour une stratégie SJF et calculer le temps moyen de séjour (temps de rotation).

**Question 4 :** Donnez l'assignation pour l'ordonnancement des ces processus pour une stratégie RR avec un quantum = 2 et calculer le temps moyen de séjour (temps de rotation).

**Question 5 :** Donnez l'assignation pour l'ordonnancement des ces processus pour une stratégie PCTER (Plus Court Temps d'Exécution Restant) / SRT (Shortest Remaining Time) et calculer le temps moyen de séjour (temps de rotation).

**Exercice 2 – FCFS, RR, SJF préemptif et non-préemptif**

Soient les différents processus suivants :

Processus	Date d'arrivée	Temps de traitement
A	0	3
B	1	6
C	4	4
D	6	2

Donnez le diagramme de Gantt pour l'exécution de ces différents processus en utilisant successivement les algorithmes FCFS, SJF sans préemption, SJF avec préemption et RR (quantum = 2 unités de temps et quantum = 1 unité de temps).

Pour chaque cas étudié, calculez :

- Temps de rotation de chaque processus et le temps de rotation moyen
- Temps d'attente de chaque processus et le temps d'attente moyen
- Débit (throughput)

### Exercice 3 – Ordonnancement à priorités

Pour les processus du tableau suivant, dessinez un schéma illustrant leur exécution, en utilisant l'ordonnancement avec priorité. Un nombre de priorité élevé correspond à une priorité plus importante.

Réalisez l'exercice dans une approche avec préemption et sans préemption. Calculez ensuite le temps de rotation de chaque processus.

Processus	Date d'arrivée	Temps de traitement	Priorité
A	0	5	4
B	2	4	2
C	2	2	6
D	4	4	3

Pour chaque cas étudié, calculez :

- Temps de rotation de chaque processus et le temps de rotation moyen
- Temps d'attente de chaque processus et le temps d'attente moyen
- Débit (throughput)

### Exercice 4 :

On considère 4 processus, A, B, C, D. On suppose que l'exécution des processus nécessite:

Pour A : 7 unités de temps CPU, 3 unités de temps d'E/S et 5 unités de temps CPU.

Pour B : 6 unités de temps CPU, 4 unités de temps d'E/S, 4 unités de temps CPU.

Pour C : 5 unités de temps CPU.

Pour D : 1 unité de temps CPU, 4 unités de temps d'E/S et 2 unités de temps CPU.

On suppose que :

A se présente en premier, à l'instant 0,

B se présente à l'instant 1,

C se présente à l'instant 9,

D se présente à l'instant 12.

Montrez comment les 4 processus vont utiliser le processeur dans chacun des cas suivants:

1) Chaque processus a son propre périphérique d'E/S et l'ordonnanceur fonctionne selon Premier Arrivée Premier Servi PAPS (sans préemption).

2) Chaque processus a son propre périphérique d'E/S et l'ordonnanceur utilise l'algorithme du tourniquet, avec un quantum de 5. Le temps de commutation est égal à 0. Donnez, dans ce cas, les temps de séjour des processus A, B, C et D.

3) Les trois processus utilisent le même périphérique d'E/S dont la file d'attente est gérée premier arrivée premier servi. L'ordonnanceur du processeur utilise l'algorithme du tourniquet, avec un quantum de 5.

Le temps de commutation est supposé égal à 0

### Exercice 5 :

Soit l'ensemble des processus prêts suivants :

Processus	Durée d'exécution	Date d'arrivée	Priorité
P1	7	0	2
P2	4	0	3
P3	6	1	1
P4	1	1	2
P5	2	1	3
P6	4	2	1
P7	1	2	2

L'algorithme d'ordonnancement considéré utilise 3 files d'attente numérotées suivant un ordre croissant des priorités : 1, 2, 3 (3 est la file la plus prioritaire...)

Chaque file est gérée par un tourniquet avec un Quantum=1

Un tourniquet non vide n'est activé que si les files plus prioritaires sont vides.

Questions :

En prenant en considération que le temps de commutation est nul,

1. Donnez le diagramme de Gantt relatif à l'exécution de ces processus
2. Calculez :
  - a. Le temps moyen de rotation
  - b. Le temps d'attente de chaque processus
  - c. Le débit
3. Un processus pourrait-il être victime d'une famine (attente infinie)? justifiez votre réponse.

### Exercice 6 : Ordonnancement des processus à date d'échéance (7points)

Soit un système d'exploitation doté d'un seul processeur et qui comprend exactement 5 processus nommés : A, B, C, D, E. Chaque processus sera activé à sa date de déclenchement (date d'arrivée). A cette date, un processus est mis alors dans la file des processus prêts pour demander l'unité centrale (CPU) pour une durée d'exécution donnée.

Nous supposons de plus qu'à chaque processus est associée une date d'échéance, avant laquelle il doit se terminer, sinon il y a « une faute temporelle ».

Hypothèse :

Les temps de commutation de contexte sont considérés comme négligeables.

Soit le scénario suivant :

Processus	Durée d'exécution	Date de déclenchement	Date d'échéance	Date de fin
A	12	1	16	
B	3	0	17	
C	2	2	16	
D	1	3	17	
E	1	4	15	

### Questions :

Pour chaque processus on vous demande de calculer la date de fin d'exécution selon la nature de l'ordonnancement du processeur et de dire s'il y a « une faute temporelle » ou non. Présenter, pour chaque cas, le chronogramme d'allocation du processeur.

1. Selon la politique de l'ancienneté (FIFO).
2. Selon la politique du tourniquet (Round Robin) avec un quantum de 2.

### Exercice 7 :

On considère une architecture monoprocesseur dans laquelle on désire exécuter l'ensemble des processus suivants :

Processus	Temps d'arrivée	Temps d'exécution total
A	0	10
B	0	6
C	1	8
D	5	4

Tout au long de cet exercice, nous considérons les hypothèses suivantes :

- Nous disposons d'un seul canal pour gérer un disque.

- L'ordre des services des requêtes sur le disque se fait selon la politique FCFS.
  - Une opération d'entrée-sortie commencée ne peut plus être préemptée.
  - Arrivé à la moitié de son exécution, chaque processus doit faire 3 unités de temps d'entrée-sortie, puis reprendre son exécution.
- A-** On considère un algorithme d'ordonnancement à priorité préemptif, en supposant que la priorité d'un processus est inversement proportionnelle à son temps d'exécution total restant. C'est-à-dire que le processus ayant le temps d'exécution total restant le plus court est le plus prioritaire.
- a. S'agit-il d'un algorithme à priorité statique ou dynamique ? Justifier.
  - b. Quel algorithme parmi ceux que vous connaissez produit un résultat équivalent à celui de cet algorithme ? Justifier votre réponse.
  - c. Dessiner le diagramme de GANTT :
    - Si le système a le choix entre plusieurs processus de même priorité, il favorise celui qui était en train de s'exécuter.
  - d. Calculer le temps de rotation moyen TRM (appelé aussi temps de traitement moyen) de cet algorithme en donnant la formule détaillée.