

Ordonnancement

Exercice 1. Questions de cours Les questions de cours sont à destinées à vous permettre de vérifier votre compréhension du cours. Elles sont à travailler à l'avance et ne seront pas traitées en TD ou TP.

1. Quels sont les critères d'évaluation d'un ordonnanceur ? Indiquez pour chaque critère comment il doit être utilisé.
2. Citez deux inconvénients de l'algorithme d'ordonnancement FIFO.
3. Comment estimer le temps restant d'un processus pour l'algorithme plus court d'abord ?
4. Pourquoi cherche-t-on à réduire le nombre de commutation de processus ?
5. Dans quel cas l'algorithme Round-Robin devient-il particulièrement efficace ?

Exercice 2. Problème de base On considère les cinq processus suivants pour lesquels nous indiquons la date d'arrivée ainsi que leur durée d'exécution estimée.

Processus	Arrivée	Durée
P1	0	7
P2	1	4
P3	1	2
P4	2	2
P5	3	1

1. Dessinez le diagramme de Gantt du résultat d'un ordonnancement FIFO non préemptif et indiquez le temps d'attente moyen.
2. Dessinez le diagramme de Gantt du résultat d'un ordonnancement *plus court d'abord* préemptif et indiquez le temps d'attente moyen.
3. Dessinez le diagramme de Gantt du résultat d'un ordonnancement *round-robin* non-préemptif avec un quantum de 3 et indiquez le temps d'attente moyen.
4. Quel est le meilleur algorithme suivant le critère du temps d'attente moyen ? Du temps d'attente min-max ?

Exercice 3. Problème de base On considère les processus suivants, définis par leur durée d'exécution estimée, leur date d'arrivée et leur priorité initiale. Les plus grandes valeurs de priorité indiquent les processus les plus prioritaires.

Processus	Arrivée	Durée	Priorité
P1	0	8	2
P2	0	1	1
P3	2	2	1
P4	4	6	4
P5	6	1	3

1. Dessinez le diagramme de Gantt correspondant au résultat d'un ordonnancement FIFO non préemptif et indiquez le temps d'attente moyen.
2. Dessinez le diagramme de Gantt correspondant au résultat d'un ordonnancement par priorité préemptif et indiquez le temps d'attente moyen.
3. Dessinez le diagramme de Gantt correspondant au résultat d'un ordonnancement plus court d'abord préemptif et indiquez le temps d'attente moyen.
4. Dessinez le diagramme de Gantt correspondant au résultat d'un ordonnancement round-robin non-préemptif, sans priorité, avec un quantum de temps fixé à 2 et indiquez le temps d'attente moyen.
5. Quel est le meilleur algorithme suivant le critère du temps d'attente moyen ? Du temps d'attente min-max ?

Exercice 4. Priorités dynamiques et E/S On considère les processus suivants pour lesquels on indique la date d'arrivée, la durée estimée, la priorité et les temps où ils vont réaliser une demande d'E/S :

Processus	Arrivée	Durée	Priorité	E/S
P1	0	10	4	3, 5, 9
P2	0	6	2	
P3	1	5	3	
P4	4	2	4	1
P5	8	5	2	
P6	11	6	3	

1. Dessinez le diagramme de Gantt représentant l'ordonnancement de ces processus dans le cas de l'algorithme plus court-d'abord préemptif en ignorant les priorités et les demandes d'E/S. Vous indiquerez aussi le temps d'attente moyen.

2. Toujours en ignorant les priorités et demandes d'E/S, dessinez le diagramme de Gantt dans le cas de l'algorithme *Round-Robin* avec un quantum de temps de 2. Vous indiquerez aussi le temps d'attente moyen.

On considère maintenant un système utilisant un ordonnancement par priorités dynamiques allant de 0 (la plus basse) à 4 (la plus haute). À chaque fois qu'un processus se voit attribuer le processeur, sa priorité baisse de 1 sans jamais descendre en dessous de 0.

L'algorithme d'ordonnancement choisit le processus dont la priorité est la plus élevée et pour un niveau de priorité il utilise l'algorithme *Round-Robin* avec un quantum de 2.

3. Dessinez le diagramme de Gantt pour ce nouveau système, toujours en ignorant les demandes d'E/S. Vous indiquerez aussi le temps d'attente moyen.

Les processus peuvent faire des demandes d'E/S qui suspendent le processus pendant 5 unités de temps. Lorsqu'un processus repasse dans la file "Prêt" après une demande d'E/S, sa priorité est réinitialisée à sa valeur initiale mais il ne peut pas préempter le processus actuellement en cours d'exécution.

4. Dessinez le diagramme de Gantt représentant l'ordonnancement des processus dans le cas de ce système avec priorité et demandes d'E/S. Vous indiquerez aussi le temps d'attente moyen.

Exercice 5. Priorités dynamiques et E/S On considère les processus suivants définis par leur date d'arrivée, leur durée estimée, leur priorité initiale ainsi que les entrées/sorties qu'ils vont effectuer. Les priorités sont comprises entre 0 et 3, les plus grandes valeurs indiquant les processus les plus prioritaires. Pour chaque E/S on indique après combien de temps d'exécution le processus réalise la demande et, entre parenthèses, combien de temps celle-ci va durer.

Processus	Arrivée	Durée	Priorité	E/S
P1	0	4	2	3(3)
P2	0	6	1	
P3	3	7	3	
P4	4	4	2	1(3), 2(4)
P5	7	1	2	

1. On ignore pour l'instant les priorités et les demandes d'E/S. Dessinez le diagramme de Gantt représentant l'ordonnancement des processus pour l'algorithme *plus court d'abord* préemptif. Vous indiquerez les temps d'attente moyen.

2. Toujours en ignorant les priorités et les demandes d'E/S, dessinez le diagramme de Gantt représentant l'ordonnancement des processus par l'algorithme *round-robin* non-préemptif avec un quantum de 2. Vous indiquerez aussi le temps d'attente moyen.

On introduit maintenant les priorités dans le système. L'ordonnancement est de type *round-robin* non-préemptif par niveau de priorité avec un quantum de 2. Lorsqu'un processus utilise l'intégralité de son quantum, sa priorité est réduite de 1. Si au moment d'ordonnancer tous les processus ont une priorité de 0, ils sont tous remontés à une priorité de 2.

3. Dessinez le diagramme de Gantt représentant l'ordonnancement selon cet algorithme et indiquez le temps d'attente moyen.

Enfin, nous introduisons la prise en compte des E/S qui suspendent le processus pour une certaine durée le temps que la demande soit satisfaite. Lorsque l'E/S se termine, le processus revient dans l'état prêt avec une priorité augmentée de 1 mais sans possibilité de préempter.

4. Dessinez le diagramme de Gantt représentant l'ordonnancement selon cet algorithme et indiquez le temps d'attente moyen et maximal.

Exercice 6. Système à deux processeurs On considère les processus suivants, définis par leur date d'arrivée et leur durée estimée :

Processus	Arrivée	Durée
P1	0	3
P2	6	1
P3	2	2
P4	1	4
P5	5	3

1. Dessinez le diagramme de Gantt représentant l'ordonnancement selon l'algorithme *plus court d'abord* préemptif et indiquez le temps d'attente moyen.
2. Dessinez le diagramme de Gantt représentant l'ordonnancement selon l'algorithme *round-robin* avec un quantum de 2 et indiquez le temps d'attente moyen.
3. Dessinez le diagramme de Gantt reprenant la question 1 sur un système disposant de deux processeurs.
4. Dessinez le diagramme de Gantt reprenant la question 2 sur un système disposant de deux processeurs.

 **Exercice 7. Cas d'étude** On considère maintenant les processus suivants. Les valeurs de priorités les plus grandes indiquent les processus les plus prioritaires, pour les E/S sont indiqué le temps (dans l'exécution du processus) où se produit l'E/S puis sa durée entre parenthèses.

Processus	Arrivée	Durée	Priorité	E/S
P1	0	10	3	4(2), 9(2)
P2	0	3	2	
P3	2	1	3	
P4	6	8	2	2(8)
P5	8	2	4	1(5)
P6	8	6	1	

Première approche : On considère le principe d'ordonnancement suivant, inspiré de l'algorithme utilisé sous Windows dans lequel chaque processus est muni d'une valeur de priorité dynamique :

- lorsqu'un processus a consommé son quantum de temps, il est interrompu et voit sa priorité diminuer de 1 (une priorité ne peut jamais être inférieure à zéro) ;
- lorsqu'un processus est interrompu avant la fin de son quantum, sa priorité ne change pas et il terminera son quantum la prochaines fois qu'il sera sélectionné par l'ordonnanceur ;
- lorsqu'un processus retourne dans la file "prêt" après avoir réalisé une E/S, il voit sa priorité augmentée de 2.

L'algorithme est préemptif : il choisit systématiquement le processus ayant la priorité la plus élevée. Au sein d'un même niveau de priorité, il utilise le principe du Round-Robin avec un quantum de 2.

1. Dessinez le diagramme de Gantt correspondant à l'exécution de ces 5 processus pour l'algorithme décrit et indiquez le temps d'attente moyen.

Deuxième approche : on utilise maintenant un autre algorithme d'ordonnancement, inspiré des algorithmes utilisées par les systèmes Linux et MacOS X, qui utilise aussi des priorités dynamiques pour les processus :

- lorsque un processus a consommé son quantum de temps, il est interrompu et voit sa priorité diminuer de 1 (une priorité ne peut jamais être inférieure à zéro) ;
- lorsque tous les processus prêts ont une priorité de 0, la priorité de chaque processus est modifiée :

$$\text{nouvelle priorité} = \frac{\text{priorité actuelle}}{2} + \text{priorité initiale.}$$

L'algorithme est préemptif : il choisit systématiquement le processus ayant priorité la plus élevée. Au sein d'un même niveau de priorité, il utilise le principe du Round-Robin avec un quantum de 2.

2. Dessiner le diagramme de Gantt correspondant à l'exécution de ces processus pour l'algorithme décrit et indiquez le temps d'attente moyen.