



Examen du 6 mars 2009 (2h)

RSA : Réseaux et Systèmes Avancés – 2A

Documents interdits, à l'exception d'une feuille A4 à rendre avec votre copie.



★ **Exercice 1.** Ordonnancement du processeur et du disque (4pts).

On considère un système monoprocesseur et les quatre processus P1, P2, P3 et P4 qui effectuent du calcul et des entrées/sorties avec un disque selon les temps donnés ci-contre. Les processus sont disponibles dès le début, dans cet ordre.

	P1	P2	P3	P4
CPU	3	2	1	4
E/S	2	4	3	2
CPU	2	3	1	
E/S	1	1		
CPU	1			

Les questions sont indépendantes. Pour chaque question, complétez les chronogrammes fournis. Chaque colonne correspond à une unité de temps. Indiquez sur le chronogramme du processeur le numéro du processus ordonnancé sur la première ligne ainsi que la file d'attente sur les lignes suivantes. Les chronogrammes disque se comprennent de façon similaire.

▷ **Question 1.** L'ordonnancement du processeur et du disque suivent la politique STF sans préemption.

Processeur	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
Disque	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>

▷ **Question 2.** L'ordonnancement du processeur et du disque suivent la politique FIFO sans préemption.

Processeur	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
Disque	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>

▷ **Question 3.** Processeur : Round Robin de quantum 2 unités de temps ; Disque : FIFO sans préemption.

Processeur	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
Disque	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>
	<table border="1" style="width: 100%; height: 15px;"></table>

★ **Exercice 5. Questions de cours** (3pts).

- ▷ **Question 1.** Qu'est ce que la TLB ? À quoi cela sert-il ?
- ▷ **Question 2.** Décrivez les workloads d'utilisation mémoire «rampe», «pic» et «plateau». Donnez des exemples de programme pouvant présenter ces caractéristiques.
- ▷ **Question 3.** Quelle est la solution classique pour donner l'impression que la machine dispose de plus de ressources physiques qu'en réalité ? Pourquoi cela fonctionne-t-il le plus souvent ?
- ▷ **Question 4.** Qu'est ce que la fragmentation interne ? Externe ?
- ▷ **Question 5.** En quoi consiste un mécanisme de priorité dynamique ? Pourquoi introduire une telle chose dans l'ordonnanceur processeur ?
- ▷ **Question 6.** Qu'est ce que le MMU ? À quoi cela sert-il ? Quels sont les avantages offerts par son utilisation ?

★ **Exercice 6. Pagination mémoire** (5pts).

On considère une mémoire segmentée paginée pour laquelle les cases en mémoire centrale sont de 1 Ko. La mémoire centrale compte au total 20 cases numérotées de 1 à 20. Dans ce contexte, on considère trois processus A, B et C. Le Processus A est composé de trois segments S1A, S2A et S3A de tailles respectives 2 Ko, 7 Ko et 4 Ko. Le processus B est composé de deux segments S1B et S2B de tailles respectives de 8 Ko et 4 Ko. Le processus C est composé d'un seul segment S1C de taille 3 Ko.

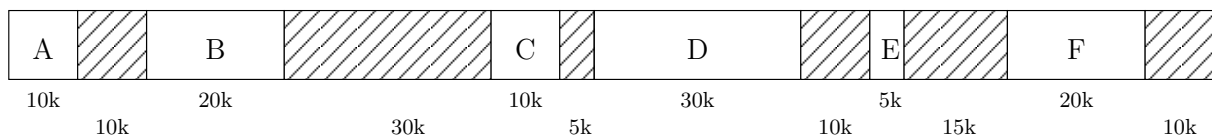
Seules les pages 1 et 2 du segment S1A, la page 2 de S2A, la page 1 de S3A, les pages 2 et 4 de S1B, la page 1 de S2B et la page 2 de S1C sont chargées en mémoire centrale respectivement dans les cases 4, 5, 10, 6, 11, 2, 15 et 12.

- ▷ **Question 1.** Représentez sur un dessin les structures allouées (table des segments, tables des pages) et la mémoire centrale correspondant à l'allocation décrite.
- ▷ **Question 2.** Donnez l'adresse physique correspondant à l'adresse virtuelle 2095 dans le processus A. Même question si la requête est faite depuis le processus B. Même question si la requête est faite depuis le processus C.
- ▷ **Question 3.** Que se passe-t-il si le processus A accède à l'adresse virtuelle 4295 ? Même question si la requête est faite depuis le processus B. Même question si la requête est faite depuis le processus C.

★ **Exercice 7. Allocation mémoire** (2pts).

On suppose que la situation initiale de la mémoire est telle que représentée sur le schéma suivant. Le workload est le suivant :

- | | |
|---------------------------------|---------------------------------|
| 1. Allocation d'un bloc G (20k) | 5. Allocation d'un bloc I (40k) |
| 2. Libération du bloc B | 6. Allocation d'un bloc J (5k) |
| 3. Allocation d'un bloc H (15k) | 7. Libération du bloc C |
| 4. Libération du bloc E | 8. Allocation d'un bloc K (15k) |



Dans chacune des deux questions suivantes, on se place dans un contexte différent. Dans chaque cas, nous allons étudier le comportement des trois algorithmes "Best Fit" et "Next Fit". Pour chaque algorithme (et donc deux fois par question), il vous est demandé d'indiquer les éventuelles allocations qui échouent et de dessiner la situation de la mémoire à la fin du workload. Justifiez correctement vos réponses.

- ▷ **Question 1.** On suppose qu'on gère une mémoire segmentée comme on en trouvait sur PDP-7.
- ▷ **Question 2.** On suppose maintenant que l'on gère la mémoire au niveau malloc. Qu'est ce que cela change par rapport à la question précédente ?