

Partiel – 18 février 2014

R. Bonaque, H. Ding, M. Gleize, N. Sabouret

L'épreuve dure 2h00. Tous les documents sont autorisés. Les exercices sont indépendants.

Exercice 1 – Question de cours (2,5 points)

1. Dessinez sur un schéma les composants matériels principaux d'un ordinateur. Les périphériques externes seront représentés par un bloc unique. (1,5 points)
2. Rajoutez un bloc pour les processus et un pour l'OS sur votre schéma puis indiquez à l'aide de flèches nommées et facilement différenciables la provenance et la destination d'une exception logicielle, d'une exception matérielle et d'un appel système. (1 point)

Exercice 2 – Ordonnancement (4 points)

On considère les processus suivants, définis par leur durée (réelle ou estimée) et leur date d'arrivée :

P1 durée : 8, date 0

P2 durée : 3, date 2

P3 durée : 4, date 4

P4 durée : 1, date 4

P5 durée : 4, date 12

1. Dessinez un diagramme de Gantt correspondant au résultat d'un ordonnancement pré-emptif plus court d'abord (avec remise en fin de file) et indiquez le temps d'attente moyen. (1,5 points)
2. Dessinez un diagramme de Gantt correspondant au résultat d'un ordonnancement « round robin » avec un quantum de temps fixé à 2 et indiquez le temps d'attente moyen. (1,5 points)
3. Quel est le meilleur algorithme suivant le critère du temps d'attente moyen ? Du temps d'attente min-max ? (1 point)

Exercice 3 – Allocation mémoire contigüe (3 points)

On se place dans un système de mémoire de 1200 Ko de mémoire haute (c'est-à-dire au delà de la partie utilisée par l'OS) en partitionnement variable (swapping).

Des processus se présentent à la mémoire au temps 0 ms dans l'ordre suivant :

- P1 : 600Ko de mémoire, durée 2ms
- P2 : 150Ko de mémoire, durée 10ms
- P3 : 400Ko de mémoire, durée 2ms
- P4 : 300Ko de mémoire, durée 3ms
- P5 : 200Ko de mémoire, durée 4ms
- P6 : 400Ko de mémoire, durée 2ms

1. Déroulez l'algorithme d'allocation First-fit (prochain bloc libre) en précisant à chaque instant les allocations et désallocations qui sont effectuées, et donnez le taux de fragmentation au temps 3 ms (1 point)
2. Déroulez l'algorithme d'allocation Best-fit (plus petit bloc libre) en précisant à chaque instant les allocations et désallocations qui sont effectuées, et donnez le taux de fragmentation au temps 3 ms (1 point)
3. Déroulez l'algorithme d'allocation Worst-fit (plus grand bloc libre) en précisant à chaque instant les allocations et désallocations qui sont effectuées, et donnez le taux de fragmentation au temps 3 ms (1 point)

Exercice 4 – Pagination (7,5 points)

1. Expliquez pourquoi les tailles de pages sont toujours une puissance de 2 (1 point)
2. On suppose un espace d'adresses logiques de 64 pages de 256 octets chacune, représenté dans une mémoire physique de 128 cadres de pages. Combien de bits comporte l'adresse logique ? L'adresse physique ? Expliquez. (1 point)
3. On suppose maintenant un système de 4096 Ko de mémoire haute organisée avec des pages de 32Ko (un seul niveau de pagination). Décrivez le système d'adressage logique. Quelle est la taille maximum de la table des pages ? Expliquez. (1 point)
4. On suppose que, dans le système de la question précédente, on a trois processus qui s'exécutent sur le système : P1 nécessitant 1250Ko (code, données et pile), P2 nécessitant 100 Ko et P3 nécessitant 200 Ko. Quelle est la quantité de mémoire réellement utilisée par l'exécution de ces trois processus ? Quel est le taux de fragmentation ? Expliquez. (2 points)
5. En considérant les huit premières entrées de la table de page présentée par la figure suivante :

N° cadre de page	N° de page	Bit de présence/absence
7	0	0
6	0	0
5	0	1
4	2	1
3	0	0
2	1	1
1	0	0
0	3	1

et en supposant une taille de cadre de page de 32Ko, donner les adresses logiques correspondantes aux adresses physiques 31792 et 90348 ? Expliquez. (2,5 points)

Exercice 5 – Estimation du temps UC (3 points)

Un certain nombre d'algorithmes d'ordonnancement ont besoin d'une estimation du temps processeur que va prendre le prochain cycle d'un processus. On supposera, comme dans le cours, que cette estimation est la moyenne exponentielle des cycles précédents, avec un facteur de pondération α . On note d_n l'estimation du temps pris par le cycle n et t_n son temps réel. On a donc (comme vu en cours) :

$$d_n = \alpha d_{n-1} + (1 - \alpha)t_{n-1}$$

1. Exprimez d_n en fonction de d_0 de α et des $(t_i)_{i \in \mathbb{N}}$. (1 point)
2. On rappelle que pour $x \in \mathbb{R}$ $\exp(x) = \sum_{k=0}^{+\infty} \frac{x^k}{k!}$. Vers quelle valeur l'expression $\frac{d_n}{\alpha^n}$ tend-elle si $d_0 = 0$ et $t_i = \frac{1}{i!}$ ms pour tout $i \geq 0$? (1 point)
3. Que se passerait-il avec ce processus si l'ordonnanceur avait comme stratégie « le plus court d'abord »? Cela vous paraît-il possible pour un processus réel, se passerait-il la même chose avec un « round-robin »? (1 point)