

## Examen [Session Principale]

Niveau d'Etude : 1<sup>ière</sup> année SIL-ARS-SE

Matière : Systèmes d'Exploitation 1

Enseignants Responsables: W. Youssef, A. Gazdar,  
A. Ben Chikha

Année Universitaire : 2010/2011

Semestre : 2<sup>ème</sup> Semestre

Date : 2 Juin 2010

Durée : 2h

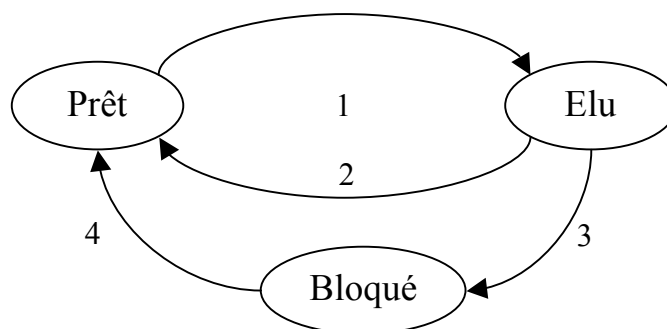
Documents : Non autorisés

Nombre de pages : 7

*Le barème est donné à titre indicatif. La clarté de la copie rendue sera prise en considération.*

### EXERCICE 1 : Questions de Compréhension (5 points : 1 x 5 )

1. Etant donné le diagramme d'états/transitions suivant, citer la/les transition(s) qui doivent être supprimée(s) si on utilise un algorithme d'ordonnancement sans réquisition (non préemptif). Justifier votre réponse.



La transition à supprimer est la transition qui passe de l'état Elu à l'état Prêt, car cette transition a uniquement lieu lors d'une préemption.

2. Qu'est-ce qu'un PCB ? Citer 3 attributs du PCB.

PCB : Process Control Bloc. C'est une structure de contrôle qui contient toutes les informations permettant de décrire le contexte d'un processus. Parmi ses attributs, on cite : PID et PPID, Etat, Priorité, Compteur Ordinal, Pointeurs, Temps d'exécution passé...

3. Qu'est ce que la MMU ? Expliquer son rôle.

MMU : Memory Management Unit. Unité de gestion de la mémoire, permettant notamment de traduire les adresses virtuelles en adresses physiques.

4. Qu'est ce que la fragmentation ? Quelle est la différence entre une fragmentation interne et externe ?

La fragmentation est le fait de créer des fragments inutilisés dans la mémoire suite à l'allocation des processus. Ces fragments sont trop petits pour être alloués. Ils peuvent se trouver à l'intérieur d'une partition (fragmentation interne), en particulier dans le cas d'une allocation contiguë à partitions fixes ; ou entre les partitions (fragmentation externe), dans le cas d'une allocation contiguë à partitions dynamiques.

5. A votre avis, où est-ce qu'on peut stocker les informations concernant les algorithmes de remplacement de page, telle que la date de chargement d'une page utilisée dans l'algorithme FIFO ?

Les informations nécessaires aux algorithmes de remplacement de page peuvent être stockées dans la table des pages.

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.

Complétez l'Annexe B, et donnez le temps de rotation moyen obtenu.

$$\text{TRM} = (15+21+16+24)/4 = 19$$

Complétez l'Annexe C, et donnez le temps de rotation moyen obtenu.

[illegible][illegible][illegible][illegible]

$$\text{TRM} = (23 + 24 + 15 + 20) = 20,5$$

4. Comparez les différents temps de rotation calculés précédemment, et interprétez le résultat.

Nous remarquons que le temps de rotation moyen de l'algorithme tourniquet est plus élevé que celui de l'algorithme de priorité. Cela est dû au fait que l'algorithme de tourniquet est équitable, et qu'il répartit le temps entre les processus, ce qui augmente leur temps d'existence dans le système.

Pour cet exemple, le changement de la politique d'ordonnancement pour les entrées/sorties n'a pas influé sur le temps de rotation moyen.

### **EXERCICE 3 : Gestion de la Mémoire (3 points)**

Soit la suite des pages suivantes  $\{ 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5 \}$ .

Donner l'évolution de la mémoire centrale en utilisant les algorithmes de remplacement de page suivants et en donnant le nombre de défauts de page:

- Pour l'algorithme FIFO avec nombre de case mémoire=3, puis avec nombre de case mémoire =4. Que constatez-vous ? Expliquez.

- Pour l'algorithme optimal avec nombre de case mémoire=4.

### Mémoire à 3 cases : FIFO

1	1	1	4	4	4	5	5	5	5	5	5
	2	2	2	1	1	1	1	1	3	3	3
		3	3	3	2	2	2	2	2	4	4

=> 9 défauts de page

### Mémoire à 4 cases : FIFO

1	1	1	1	1	1	5	5	5	5	4	4
	2	2	2	2	2	2	1	1	1	1	5
		3	3	3	3	3	3	2	2	2	2
			4	4	4	4	4	4	3	3	3

=> 10 défauts de page

On remarque que l'augmentation du nombre de cases mémoires n'entraîne pas nécessairement la diminution du nombre de défauts de page. Dans ce cas, le nombre de défauts de pages augmente, principalement parce que le nombre de défauts de pages initiaux (par défaut) augmente.

### Mémoire à 4 cases : Optimal

1	1	1	1	1	1	1	1	1	1	4	4
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3
			4	4	4	5	5	5	5	5	5

=> 6 défauts de page

## EXERCICE 4 : Gestion de la Mémoire (5 points : 1 x 5)

Etant donnés :

- Une table des pages de taille 128 Ko
- Le nombre d'entrées de la table des pages égal à 65536
- Le déplacement est codé sur 16 bits

- Une entrée de la table des pages est de la forme :  $|n|1|3|$  où
    - $n$  est nombre de bits pour coder un cadre de page (une case)
    - $1$  est le bit d'absence/présence
    - $3$  est le nombre de bits pour coder la date de chargement de la page
- ➔ Chaque entrée de la table contient donc  $n+4$  bits

On vous demande de (vous devez détailler votre réponse) :

1- Déterminer la valeur de  $n$ .

Taille de la TP = taille d'une entrée de la TP \* nb entrées dans la TP

$$128 \text{ Ko} = (n+4) * 2^{16}$$

$$\Rightarrow n = [(2^7 * 2^{10} * 2^3) / 2^{16}] - 4 = 2^4 - 4 = \mathbf{12}$$

2- Calculer la taille de la mémoire physique.

Taille MP = nb cadres \* taille d'un cadre

$$= 2^n * 2^{\text{dep}}$$

$$= 2^{12} * 2^{16} = 2^{28} \text{ octets} = \mathbf{256 \text{ Mo}}$$

3- Donner la taille de l'espace d'adressage virtuel

Taille ML = nb pages \* taille d'une page

$$= \text{nb entrées dans la TP} * 2^{\text{dep}}$$

$$= 2^{16} * 2^{16} = 2^{32} \text{ octets} = \mathbf{4 \text{ Go}}$$

4- Donner le nombre de bits du bus d'adressage.

Nbre de bits du bus d'adressage = 32 bits

5- Dire si le nombre d'entrées de la table des pages change si on augmente la taille de la mémoire physique de 1 Mo. Si oui, de combien augmente-il ?

Le nombre d'entrées de la TP n'est pas influencé par la taille de la mémoire physique, car il représente le nombre de pages, il dépend donc de la taille de l'espace d'adressage virtuel (mémoire logique).

BON TRAVAIL.