Systèmes d'exploitation

Exercices accompagnant le cours : Mémoire (partie 2)

Objectifs des exercices

Comprendre l'organisation de la mémoire virtuelle, la pagination et la segmentation

1 Quelques calculs

1.1 Exercice 1

Soit un ordinateur avec un processeur multi-cœurs. Le processeur contient 4 cœurs 64 bits. La taille de la mémoire physique est de 4 Go. La taille d'un cadre (page physique) est égale à celle d'une page mémoire, qui est de 8 ko. Il y a une seule mémoire centrale dans le système. La taille du fichier d'échange (swap) est de 6 Go.

1. Quelle est la taille de l'espace virtuel adressable par le processeur multi-cœurs ?
2. Quel est le nombre de cadres physiques disponibles ?
3. Quel est le nombre total de pages de la mémoire virtuelle pouvant être allouées sur le système ?

Imaginons un processeur 64 bits réservant 40 bits de l'adresse virtuelle pour définir le numéro de page.

1. Indiquez la taille maximale d'une page

2. Indiquez le nombre maximum de page

2 Pagination

En cas de défaut de page, le système d'exploitation est chargé de rechercher un cadre disponible en mémoire physique afin de pouvoir y placer la page. Cependant, au cours d'une utilisation plus ou moins prolongée, il peut arriver que la mémoire physique soit saturée. Dans ce cas un algorithme de pagination a la responsabilité de choisir une page *victime*.

2.1 Exercice 1

Supposons que nous ayons une mémoire physique de 3 cadres. Étant donné la séquence d'accès aux pages suivantes (pour un processus donné) :

Séquence	8	0	1	2	1	0	1	2	5	2	1	0	1	4	1	2	1	0	5	2	1	8

Donnez le déroulement du remplissage des pages et le nombre de défauts de pages pour les politiques de remplacement suivantes :

— First In First Out (FIFO)
— Least Recently Used (LRU)
— Least Frequently Used (LFU)
— L'algorithme de remplacement de pages optimal

2.2 Exercice 2

Chacun des algorithmes de remplacement offre un niveau de performance différent. Ce niveau de performance dépend fortement de la séquence d'accès et des propriétés de cette séquence. Intuitivement, une solution permettant de réduire le nombre de défauts de pages pourrait consister à augmenter le nombre de cadres, i.e. la taille de la mémoire physique. Comparez le nombre de défauts de pages obtenu avec 3 puis 4 cadres avec l'algorithme FIFO et la séquence de référence :

Séquence	3	2	1	0	3	2	4	3	2	1	0	4

Qu'en pensez vous ?

3 Segmentation

3.1 Exercice 1

On considère la table des segments suivante pour un processus P1 :

Segment	**Base**	**Taille**
0	540	234
1	1254	128
2	54	328
3	2048	1024
4	976	200

1. Calculez les adresses réelles correspondant aux adresses virtuelles suivantes (vous signalerez éventuellement les erreurs de segmentation) :
   — (0 :128)
   — (1 :100)
   — (2 :465)
   — (3 :888)
   — (4 :100)
   — (4 :344)
2. L'adresse virtuelle (2 :328) est-elle valide ?

3.2 Exercice 2

Considérons un programme avec deux segments ; les instructions sont dans le segment 0 (droits rx) et les données dans le segment 1 (droits rw). La mémoire est paginée avec des adresses virtuelles dont le numéro de page est sur 4 bits et le déplacement sur 10 bits. Nous avons actuellement la configuration suivante :

Segment 0		Segment 1	
Page	**Cadre**	**Page**	**Cadre**
0	2	0	Sur disque
1	Sur disque	1	14
2	11	2	9
3	5	3	6
4	Sur disque	4	Sur disque
		5	13
		6	8
		7	12

Donnez l'adresse réelle qui résulte de la traduction dynamique ou identifiez le type de défaut dans chacun des cas suivants :

1. Chargement depuis le segment 1, page 1, déplacement 3
2. Rangement à l'emplacement segment 0, page 0, déplacement 16
3. Chargement depuis le segment 1, page 4, déplacement 28
4. Saut à l'emplacement segment 1, page 3, déplacement 32

Donnez l'adresse réelle correspondant aux adresses virtuelles ou identifiez le type de défaut dans chacun des cas suivants :

1. Chargement depuis le segment 1, adresse 5013
2. Saut à l'emplacement segment 0, adresse 5120