

TD 7 – Remplacement de pages

Attention : Dans ce TD, toutes les *adresses* et *portions d'adresses* sont donnés en hexadécimal.

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. Donnez les deux grandes classes d'algorithmes de remplacement de pages.
2. En pratique, quel est le meilleur algorithme ? Pourquoi ?
3. Quel est l'inconvénient de l'algorithme FIFO avec seconde chance (FIFO-2) ?
4. Quel est la particularité de l'algorithme NRU ?
5. Quel est l'inconvénient de l'algorithme NRU ?

Exercice 2. Segmentation paginée

On considère un système muni de 64 ko de mémoire physique gérée de manière segmentée et paginée. Chaque processus peut utiliser 16 segments de 1 ko et le système supporte jusqu'à 256 processus. Les cadres de page font 512 o.

1. Quelle est la taille de l'adresse physique ?
2. Quelle est la taille de l'adresse logique ?
3. Quelle est la taille maximale de la mémoire virtuelle ?
4. Rappeler ce qu'est un segment global.
5. Dans cette question, on suppose que la moitié des segments sont globaux. Dans ce contexte, quelle est la taille de la mémoire virtuelle ?
6. Combien de pages un processus peut-il utiliser au maximum ?
7. On rappelle que l'adresse linéaire doit permettre d'adresser toute la mémoire d'un processus, mais qu'elle n'a pas besoin d'adresser l'ensemble de la mémoire virtuelle. Quelle est la taille et la composition de l'adresse linéaire ?
8. Quelle est la taille (en nombre de bits) de chaque ligne de la table des descripteurs lorsque tous les segments sont locaux ?
9. Même question si la moitié des segments sont globaux (ce qui est la configuration habituelle sur un OS moderne).

À un moment de l'exécution, plusieurs processus P_1, P_2, \dots, P_N sont en exécution dans le système. L'état du système est partiellement décrit ci-après :

Table des segments de P_1 :

Segment	Limit	Base
00	085	0000
01	3B6	3000
02	341	2000
03	225	0086
04	05F	1B80

Table des segments de P_2 :

Segment	Limit	Base
00	0A5	1001
01	107	0C00
02	3B6	3000
03	0A3	01CF

Table des pages de P_1 :

Page	Cadre	Valide
00	36	0
01	7A	0
02	32	1
0E	00	1
0F	2A	0
10	2B	0
11	14	1
18	6C	1
19	55	1
1A	31	1
1B	30	0

Table des pages de P_2 :

Page	Cadre	Valide
00	24	0
01	32	0
08	2A	1
09	76	1
0A	54	0
18	6C	1
19	55	1
1A	31	1

10. Quelle est l'adresse physique qui correspondant à l'adresse logique 0B50 pour le processus P_1 ?

11. Quelle est l'adresse physique correspondant à l'adresse logique 0B50 pour le processus P_2 ?

12. Quelle est l'adresse physique correspondant à l'adresse logique 0C0D pour le processus P_1 ?

Exercice 3. Gestion de la mémoire

On considère un système de gestion de mémoire de 64Mo de mémoire physique, gérée de manière segmentée et paginée avec deux niveaux de pagination. Un processus peut avoir au plus 256 segments. Chaque segment peut adresser au plus 64ko de mémoire. Enfin, la taille des cadres de page est fixée à 4ko et le répertoire contient 4 lignes.

1. Quelle est la taille et la composition de l'adresse logique ? Indiquez sa composition.

2. Quelle est la taille et la composition de l'adresse linéaire ?

3. Quelle est la taille et la composition de l'adresse physique ?

Soit un processus muni de la table des segments et du répertoire suivant :

Segments :

Segment	Base	Limite
00	BE 0A 75	05 00
01	BE 23 D1	00 BF
02	BE 00 DA	03 61
03	BE 1A 26	05 D2
04	BE 0F F0	00 CF

Répertoire :

Répertoire	Table	Valide
0	0	1
1	3	0
2	1	1
3	2	0

et de deux tables de pages (on ne fait figurer ici que quelques pages, les autres ont leur bit de validité à 0) :

Table 0 :

Page	Cadre	Valide
2A0	23 40	1
2A1	05 BB	1
2A2	00 00	0
2A3	14 E0	1

Table 1 :

Page	Cadre	Valide
3E0	00 00	0
3E1	27 FD	1
3E2	00 00	0
3E3	3A F6	1

4. Quelle est l'adresse physique correspondant à l'adresse logique 03 00 F0 ? Indiquez clairement les valeurs calculées pendant le processus de traduction.

5. Quelle est la quantité de mémoire physique utilisée par le processus ? Justifiez.

Exercice 4. Remplacement de page

Durant son exécution, un programme accède successivement à la liste suivante de pages virtuelles de son espace d'adressage :

0, 1, 4, 2, 0, 1, 3, 0, 1, 4, 2, 3

On suppose que le système d'exploitation a alloué 3 cadres de pages au processus.

1. Donnez la suite des pages présentes en mémoire ainsi que le nombre de défauts de page pour les algorithmes suivants : FIFO simple, FIFO avec seconde chance et LRU.
2. Quel serait le remplacement optimal (si on pouvait deviner à l'avance les appels que va faire le programme) et donc le taux de défaut de page minimum ?