

TD 6 – Segmentation

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. En quoi consiste la pagination de la mémoire ?
2. Quel est le principal problème de la pagination ?
3. Quel est l'intérêt de la pagination hiérarchique ?
4. Donnez deux avantages de la mémoire virtuelle.

Exercice 2. Fondements

On suppose un espace d'adresses logiques de 8 pages de 1024 octets chacune permettant d'accéder à une mémoire physique de 32 cadres de pages.

1. Expliquez pourquoi les tailles de pages sont toujours une puissance de 2.
2. Combien de bits comporte l'adresse logique ? L'adresse physique ?
3. On suppose maintenant que le système dispose de 2048ko de mémoire logique organisé avec des pages de 8ko. Décrivez le système d'adressage logique.
4. Quelle est la taille maximum de la table des pages ?
5. On suppose que, dans le système de la question précédente, on a trois processus qui s'exécutent sur le système : P1 nécessitant 200ko, P2 de 545ko et P3 de 337ko. Quelle est la quantité de mémoire réellement utilisée par l'exécution de ces trois processus ? Quel est le taux de fragmentation ?

Exercice 3. Pagination à 1 niveau

On considère un système 32bits utilisant la technique de pagination avec des cadres de pages de 64ko. Chaque processus peut utiliser au plus 4Go de mémoire virtuelle et le système peut accueillir jusqu'à 2048 processus.

1. Quelle est la taille maximale de la mémoire physique ?
2. Quelle est la taille de l'adresse logique ?
3. Quelle est la quantité maximale de mémoire virtuelle utilisée par le système ?
4. On suppose que le système dispose de 1Go de mémoire physique. Quelle est la taille de l'adresse physique ?
5. Sur combien de bits est codé le décalage dans l'adresse logique ? Dans l'adresse physique ?
6. Sur combien de bits est codé le numéro de page dans l'adresse logique ?
7. Sur combien de bits est codé le numéro de cadre dans l'adresse physique ?
8. Quelle est la taille maximale de la table des pages d'un processus ?
9. En considérant les huit premières entrées de la table de page donnée ci-après, calculez les adresses physiques correspondant aux adresses logiques 00030B72 et 00060D81 ?

Page	Cadre	Validité
0	000	0
1	010	0
2	000	1
3	0B3	1
4	2A0	0
5	09B	0
6	0F0	1
7	2DD	1

10. À l'inverse, pouvez-vous indiquer si la donnée située à l'adresse physique 2A0DE37 appartient au processus ? Expliquez.

11. Est-ce que 37 est une adresse physique valide ? Et 90003145 ? Expliquez.

Exercice 4. Pagination à 1 niveau

On considère un système disposant de 4Mo de mémoire physique utilisant la technique de la pagination à 1 niveau avec des cadres de page de 1ko. Chaque processus dispose de 16Mo de mémoire virtuelle.

1. Quelle est la taille des adresses physiques ?
2. Sur combien de bits est codé le numéro de page dans l'adresse physique ?
3. Quelle est la taille des adresses logiques ?
4. Combien de lignes y a-t-il dans la table des pages d'un processus ?

Le tableau suivant donne un extrait de la table des pages d'un processus :

Page	Cadre	Validité
AC2	000	0
AC3	35E	1
AC4	12A	1
AC5	000	0
AC6	000	0
AC7	000	0
AC8	000	0
AC9	1A7	0

5. Calculez les adresses physiques correspondant aux adresses logiques 2B0E4C et 2B27F0.
6. Calculez l'adresse logique que le processus doit utiliser afin de réaliser un accès mémoire à l'adresse physique 4A8F1.
7. Que doit faire l'OS pour que le processus puisse réaliser un accès mémoire à l'adresse physique D12A ?

Exercice 5. Pagination à 2 niveaux

On se place dans un système de mémoire de 4Go de mémoire géré de manière paginée avec des cadres de page de 4Ko. Chaque processus peut utiliser jusqu'à 64Mo de mémoire. Le système d'exploitation autorise jusqu'à 1024 processus.

1. Quelle est la taille (en bits) de l'adresse logique
2. Quelle est la taille (en bits) de l'adresse physique
3. Combien y a-t-il de cadres de page dans la RAM ?
4. Combien chaque processus peut-il contenir de pages ?
5. On suppose que la pagination se fait sur un seul niveau. Quelle quantité de mémoire est consommée par les tables des pages ?
6. On suppose que la pagination se fait sur deux niveaux avec un répertoire sur 6 bits. Quelle quantité de mémoire est consommée par un processus qui utilise les plages d'adresses logiques suivantes ?
 - de 0 00 1B 07 à 0 00 2C 08
 - de 0 03 CA 00 à 0 0E 00 00
 - de 0 0D AB 10 à 0 0F B1 CC
 - de 0 2B 10 21 à 0 2B 3A 72

Exercice 6. Pagination à 2 niveaux

On considère un système de gestion de mémoire paginée à deux niveaux tel que :

- Les adresses virtuelles et physiques sont toutes deux codées sur 32bits ;
- Les 10 premiers bits de l'adresse virtuelle forment le premier index ($n_1 = 10$), les 10 bits suivant forment le deuxième index ($n_2 = 10$) et les 12 bits restant le déplacement ($m = 12$) ;
- Les tables de pages et le répertoire de chaque processus sont stockées dans la RAM et chaque entrée correspond à une adresse différente dans la RAM.

1. Sur combien d'octets seront codées les données du répertoire ? Quelle est la taille maximum du répertoire ?
2. Sur combien d'octets seront codées les données d'une table de page ? Quelle est la taille maximum d'une table de pages ?

3. Quelle est la quantité maximale de mémoire nécessaire pour stocker toutes ces tables ?
4. On considère un processus avec le répertoire suivant:

Index	Table
0	0A7C 0002
1	0A7C 006E
2	0A7C 0072
3	0A7C 005A

Et on souhaite résoudre l'adresse logique: 00C02DF6. Quelle table est concernée ?

5. On suppose que la table des pages concernée (par le résultat de la question précédente) commence par les entrées suivantes:

Page	Cadre	Validité
0	17AC0	0
1	02BCF	1
2	07F93	1
3	15BB0	1

Donnez l'adresse physique correspondant à l'adresse logique 00C0 2DF6.



Exercice 7. Pagination Sv32 des processeur RISC-V

L'architecture de processeur RISC-V propose différents modes de gestion de la mémoire en fonction des besoins de l'utilisateur. Lorsque les besoins en mémoire sont limités, le mode le plus simple Sv32 est utilisé. Dans ce mode les adresses logiques sont codées sur 32bits, les adresses physiques sur 34bits et une pagination à deux niveaux est utilisée avec des pages standard de 4ko. Le répertoire et les tables de pages contiennent le même nombre de lignes.

1. Quelle est la composition des adresses logiques ?
2. Quelle est la composition des adresses physiques ?

Le répertoire et la table des pages ont la même structure, chaque ligne est composée d'un numéro de cadre sur 22bits et de 10bits d'informations supplémentaires parmi lesquels des bits de protection R, W et X indiquant si la ligne peut être utilisée pour un accès en lecture, écriture ou exécution. Pour le répertoire, ces trois bits sont normalement à zéro.

3. Quelle est la taille du répertoire et d'une table des pages ?

4. Il n'y a que 22bits disponibles dans le répertoire pour indiquer la position en mémoire physique de la table des pages. Pourquoi n'y en a-t-il pas 34 ?

Chaque ligne du répertoire dispose des bits R, W et X qui n'ont pas vraiment de sens si l'on considère qu'une ligne indique seulement où se trouve la table de pages à consulter. Si par contre on considère que le répertoire découpe la mémoire logique en $2^{10} = 1024$ zones de 4Mo qui sont ensuite découpées en $2^{10} = 1024$ pages de 4ko par les tables de pages correspondantes, on peut améliorer la gestion de la mémoire d'une manière intéressante.

Une ligne du répertoire ou les trois bits R, W et X ont pour valeur 0, indiquent que le numéro de cadre pointe vers une table des pages gérant cette zone de 4Mo sous la forme de petites pages de 4ko. Une autre combinaison de ces trois bits indique que cette zone est gérée sous la forme d'une seule grosse page de 4Mo dont l'adresse physique est donnée par le numéro de cadre.

Cette technique permet d'allouer de gros blocs de mémoire contiguë sans avoir besoin d'utiliser de tables de pages ce qui économise de la mémoire.

5. Quelle quantité de mémoire est économisée par l'utilisation d'une "superpage" par rapport à une table de pages normale ?
6. Quel autre avantage présente l'utilisation de ces "superpages" ?

L'architecture RISC-V propose d'autres modes de gestion de la mémoire allant jusqu'à 57bits d'adresses logiques pour 66bits d'adresses physiques avec 5 niveaux de pagination. Chaque niveau peut indiquer soit une table de plus bas niveau, soit une superpage de grande taille. Pour ces modes permettant de gérer de très grandes quantités de mémoire, l'utilisation des superpages devient critique pour gérer efficacement la mémoire.