

Fiche de TD3

Exercice 1. Expliquez un avantage majeur que procure l'utilisation de la mémoire virtuelle sur la performance du système.

2. Quelle est la différence principale entre un algorithme de remplacement statique de pages et un algorithme de remplacement dynamique de pages ?

Exercice 2 : Soit une mémoire centrale utilisant la technique d'allocation contiguë par partitionnement fixe. Cette mémoire est composée de 5 partitions P1, P2, P3, P4 et P5. Ces partitions ont pour tailles respectives 100, 500, 200, 300 et 600 KO. Soient 4 processus A, B, C et D de tailles respectives 212, 417, 112 et 426 KO.

Donner les différents états de la mémoire centrale pour charger les 4 processus A, B, C et D (dans l'ordre) en utilisant les algorithmes d'allocation suivants :

- a- FIRST-FIT
- b- BEST-FIT
- c- WORST-FIT
- d- NEXT-FIT

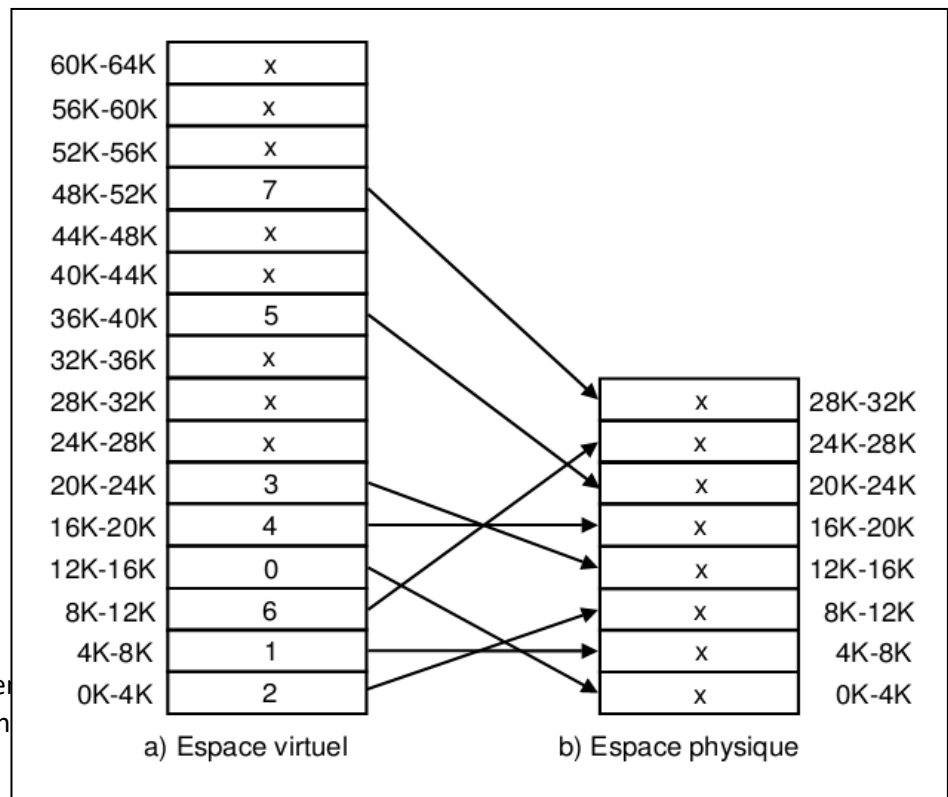
Exercice 3 : Soit la mémoire physique est de 32 Ko et l'adressage est codé sur 16 bits, l'espace d'adressage logique peut atteindre la taille 2^{16} . La taille des pages est de 4Ko.

- a. Combien de pages dispose-t-on dans l'espace d'adressage virtuel ?
- b. Combien de case dispose-t-on dans l'espace d'adressage physique ?

Considérons la représentation suivante :

- a. Donner les adresses physiques des adresses virtuelles suivantes : Adresse 0, Adresse 100, Adresse 12288 et Adresse 4097
- b. Donner les adresses virtuelles correspondant aux adresses physiques : Adresse 0, Adresse 100, Adresse 8192 et Adresse 4097

processus. Montrez la séquen



énumérés ci-dessous. Aussi, calculez le taux de faute de pages produit par chacun des algorithmes de remplacement de pages de mémoire.

1. L'algorithme optimal
2. L'algorithme du non récemment utilisé (NRU).
3. L'algorithme du moins récemment utilisé (LRU).
4. L'algorithme du premier-arrivé-premier-sorti (PAPS).

Exercice 5. Considérez une mémoire virtuelle avec une taille de mémoire physique (principale) de 1 Mo, et supportant des blocs de 128 octets. Aussi, supposez un processus occupant un espace d'adresse logique de 22 Ko.

2. Calculez le nombre de cadres dans l'espace d'adresse physique et le nombre de pages dans l'espace d'adresse logique.
3. Montrez les formats des adresses physique et logique, soit le nombre de bits pour les blocs, les cadres et les pages.
4. Déterminez l'adresse physique dans laquelle se situe l'adresse logique 10237, si l'on suppose que la page contenant l'adresse 10237 se trouve dans le cadre 1839.

Exercice 6. On considère un système de fichiers tel que l'information concernant les blocs de données de chaque fichier est donc accessible à partir de l'i-noeud de celui-ci (comme dans Unix).

On supposera que :

- Le système de fichiers utilise des blocs de données de taille fixe 1K (1024 octets).
- L'i-noeud de chaque fichier (ou répertoire) contient 12 pointeurs directs sur des blocs de données, 1 pointeur indirect simple, 1 pointeur indirect double et 1 pointeur indirect triple.
- Chaque pointeur (numéro de bloc) est représenté sur 4 octets.

(a) Quelle est la plus grande taille de fichier que ce système de fichiers peut supporter ?

(b) On considère un fichier contenant 100,000 octets. Combien de blocs de données sont-ils nécessaires au total pour représenter ce fichier sur disque ?

Exercice 7 : On considère un système disposant de 16 Ko de mémoire centrale dont 4 Ko sont occupés par la partie résidente du système d'exploitation.

Nous supposons que :

- La mémoire centrale utilise un partitionnement variable,
- La mémoire centrale utilise une stratégie de placement se basant sur l'algorithme premier ajustement ou First Fit,
- L'ordonnancement sur la mémoire se fait selon FIFO,
- L'ordonnancement sur la CPU se fait selon l'algorithme du tourniquet (Round Robin) avec un quantum $q = 3\text{ms}$,
- Chaque processus effectue un calcul suivi d'une E/S en utilisant son propre périphérique,
- Les processus arrivent conformément au tableau suivant :

Processus	Instant (t)	Taille (Ko)	Temps CPU (ms)	Durée E/S (ms)
A	0	3	9	2
B	4	5	6	9
C	6	5	4	4
D	8	4	2	6
E	10	1	4	3
F	12	1	5	1
G	16	1	3	2
H	18	3	3	8

Donnez le diagramme de GANT ainsi que les états d'occupation de la mémoire aux différentes étapes de traitement de ces processus.

Exercice 8 : On considère une machine avec quatre cases. On donne dans le tableau ci-dessous le numéro avec le moment du chargement, le temps du dernier accès, les bits R et M pour chaque page (les temps sont en tops d'horloges), ainsi que le nombre d'utilisation depuis le chargement.

Page	Chargement	Dernière référence	R	M	Nb utilisation
0	126	280	0	0	3
1	230	265	0	1	2
2	140	270	0	0	1
3	110	285	1	1	3

Si un défaut de page se produit, indiquer la page qui sera remplacée suivant l'algorithme :

- FIFO
- LRU
- LFU
- Seconde chance

Exercice 9 : On considère un système de va-et-vient avec une mémoire constituée de zones vides dans l'ordre suivant : $z_1=5Ko$, $z_2=12Ko$, $z_3=9Ko$, $z_4=3Ko$ et $z_5=10Ko$. Lorsqu'une zone libre z_i (i entier allant de 1 à 6) est plus grande que nécessaire, on la découpe en deux zones pour libérer la zone non occupée et on appelle z_{i1} la première et z_{i2} la deuxième partie (nouvelle zone libre), z_{i3} , si elle est redécoupée par la suite. Indiquer quelles zones (à l'aide de z_i , z_{i1} , z_{i2} , z_{i3}) seront prises lors des requêtes de segments successives suivantes, selon les algorithmes précisés dans le tableau :

Requêtes (dans l'ordre ci-dessous)	Algorithme First fit	Algorithme next fit	Algorithme Best fit	Algorithme Worst fit
6Ko				
4ko				
8Ko				
6Ko				

Exercice 10 : On considère les adresses virtuelles en numérotation décimale suivantes : 20000, 327688 et 64096. Pour chacune d'elles donner le numéro de page virtuelle et le déplacement.

- pour des pages de 4Ko,
- pour des pages de 8Ko.