

TD n° 4 : Gestion de la mémoire

Exercice 1

1. Quelles sont les contraintes et les problèmes que le gestionnaire de la mémoire doit résoudre ?
2. Expliquer les termes suivants :
 - a. Fragmentation.
 - b. Défragmentation.
 - c. Va et vient.
 - d. Swapping.
3. Rappeler le principe de gestion de la mémoire par :
 - a. Partitions fixes.
 - b. Partitions variables.
4. La quelle de ces deux techniques génère de la fragmentation ? Peut-on faire de la défragmentation ?

Exercice 2

1. Que signifie le mode d'allocation contiguë ?
2. Que signifie le mode d'allocation non contiguë ?
3. Qu'est-ce qu'un bloc ?
4. Comment est fixée la taille d'un bloc ?

Exercice 3

Considérons un système de gestion de mémoire à partition variables avec la liste des zones libres suivantes, ordonnée par adresse croissantes :

10Ko, 4Ko, 20Ko, 18Ko, 7Ko, 9Ko, 12Ko et 15Ko

Soit les trois demandes successives d'espace mémoire suivantes :

- a) 12Ko b) 10Ko c) 9Ko

Indiquez pour les stratégies Best-Fit, First-Fit et Worst-Fit, l'état de la mémoire après l'allocation des espaces pour les demandes de a), b) et c).

Exercice 4

On considère un espace mémoire de 20 blocs, initialement libres. La stratégie de placement est « meilleur ajustement ». Décrire l'évolution de la liste des zones libres lorsque les opérations suivantes sont effectuées :

- Allocation d'une zone de 3 blocs,
- Allocation d'une zone de 4 blocs,
- Allocation d'une zone de 2 blocs,
- Libération de la zone de 4 blocs,
- Allocation d'une zone de 1 bloc,
- Allocation d'une zone de 2 blocs,
- Libération de la zone de 3 blocs,
- Allocation d'une zone de 2 blocs.

Exercice 5

Le SE maintient en permanence une liste de zones libres : au départ, elle contient une seule zone formée de l'espace tout entier. Quand une zone est allouée, elle est prélevée à l'extrémité d'une zone libre, ce qui conduit à diminuer cette zone ou, le cas échéant, à la supprimer de la liste. La fig.1, représente un espace mémoire de 20 blocs après les allocations successives, à partir de l'état initial, de zones de tailles respectives, comptées en nombre de blocs, 3, 4, 2, 3, 2, 1 et 3.

3	4	2	3	2	1	3	libre

Fig. 1 : Allocations initiales

1. Donner le nombre de blocs libre après la libération de la 2^e, la 4^e et la 5^e zone.
2. Quel est le problème essentiel occasionné par cette organisation ?

3. Que signifie fragmentation interne ?
4. Que signifie fragmentation externe ?
5. Suite à la libération des blocs (de la question 1), comment peut-on satisfaire la demande de 5 blocs ?
6. Par quel procédé, on pourra satisfaire une requête de 11 blocs ?
7. Donner les inconvénients du compactage.
8. Que signifie « stratégie de placement » ?
9. Quels sont les critères d'appréciation de ces méthodes ?
10. Après la libération des blocs de la 1ère question, et pour satisfaire une demande de 1 bloc, préciser les zones libres choisis dans le cas où on applique :
 - a) La stratégie de 1ère zone libre (First-Fit).
 - b) Le meilleur ajustement (Best-Fit).
 - c) Le plus grand résidu (Worst-Fit).

Exercice 6

On considère la suite de demandes d'allocation (+) et de libération (-) suivantes, dans un espace mémoire de 1000 blocs, utilisant l'allocation contiguë :

+300, +200, +260, -200, +100, -300, +250, +400, -260, +150, +120, -100, -120, +200, -150, -250, +100, -400, -100, -200

Indiquer comment, à partir d'une mémoire initialement libre, le SE réalise l'allocation avec les stratégies du meilleur ajustement, de la première zone libre et du plus grand résidu.

Exercice 7

On considère une mémoire centrale où la zone « utilisateur » est structurée selon une technique de partitionnement basée sur une décomposition en puissance de 2. L'état initial de cette mémoire est composé d'une seule zone ayant une certaine taille initiale appelée **Tinit**. On suppose que cette taille est égale à **2K mots**.

À partir de cet état initial, on commence à faire des allocations et des libérations selon la technique suivante :

- (a) Les demandes d'allocation et de libération sont exprimées uniquement en puissance de 2 (avec une puissance au maximum

égale à **k : 2^k** est la taille maximale de la mémoire centrale).

- (b) Soit **T₀** la taille d'une zone **Z₀** qui est libre en mémoire centrale et soit **T_p** la taille d'un processus **P** à charger en mémoire centrale.
- (c) Deux cas sont envisageables pour charger P en mémoire centrale :
 1. Cas 1 : **T₀ = T_p**. Dans ce cas la zone **Z₀** est allouée au processus **P**.
 2. Cas 2 : **T_p < T₀**. Dans ce cas on décompose la zone **Z₀** de taille **T₀** en deux zones **Z₀₁** et **Z₀₂** ayant pour tailles respectives **T₀₁=T₀/2** et **T₀₂=T₀/2**. Donc, nous obtenons deux zones de tailles identiques (décomposition par puissance de 2).
- (d) On compare maintenant **T_p** avec **T₀₁** ou avec **T₀₂**. Si **T_p** est égale à **T₀₁** (ou égale à **T₀₂**), on alloue la zone **Z₀₁** (ou la zone **Z₀₂**) au processus **P**. Si **T_p** est toujours inférieure à **T₀₁** (ou à **T₀₂**), alors on continue la décomposition (de la zone **Z₀₁** ou bien de la zone **Z₀₂**) jusqu'à trouver une zone **Z_{jk}** dont la taille **T_{jk}** est égale à celle du processus **P**, c'est-à-dire **T_p = T_{jk}**.

Remarque : Cette décomposition peut être schématisée par un arbre binaire.

En utilisant cette technique, on vous demande de réaliser les allocations suivantes sachant que la taille initiale de la zone utilisateur est égale à 2¹² mots.

Processus	Taille demandée (en mots)
P1	2048
P2	256
P3	64
P4	32
P5	16
P6	8
P7	1024

1. On vous demande de montrer les allocations par représentation

schématique de la mémoire centrale à chaque état (depuis l'état initial jusqu'à la dernière allocation). Montrer les zones qui seront allouées aux différents processus. L'ordre des demandes est : 1, 2, 3, 4, 5, 6 et 7.

2. Est-ce que cette technique crée de la fragmentation interne, de la fragmentation externe ou les deux en même temps ? Justifier vos réponses.
3. On désire maintenant libérer l'espace détenu par les processus 3, 4, 5 et 6. Quels sont les états de la mémoire après chaque libération (ordre de libération : 6 puis 5 puis 4 et enfin 3) ?
4. Peut-on utiliser du compactage avec cette technique ? Si non pourquoi; si oui proposez une technique de compactage.
5. Que se passe-t-il si les tailles des processus ne sont plus exprimées en puissance de 2 ? Qu'est-ce qui changera par rapport à la méthode utilisant des puissances de 2.

- Un ordonnanceur (CPU scheduler) de type Round Robin avec un quantum = 2s.

Représenter, par une liste chaînée, les différentes partitions que prend la mémoire centrale au cours de l'exécution de ces cinq processus.

Exercice 8

Les processus à exécuter sont les suivants :

Processus	Taille	Temps de création	Temps d'exécution
P1	60K	0	6s
P2	100K	1	5s
P3	30K	3	2s
P4	70K	7	8s
P5	50K	9	4s

On dispose d'une machine ayant les « composantes » matérielles et logicielles suivantes :

- Une mémoire de 256K partitionnée dynamiquement. La recherche d'une partition libre se fait selon la stratégie First-Fit,
- Un moniteur (système d'exploitation) dont la taille est de 40K,