

GESTION MEMOIRE

On rappelle que les supports de cours sont disponibles à <http://mathieu.delalandre.free.fr/teachings/operating2.html>

1. Partitionnement fixe

1.1.Partitionnement fixe à taille égale

On considère une mémoire de 512 Mo gérée en partitionnement fixe à taille égale (4 partitions de 128 Mo). Au sein de ce système, l'allocation est traitée par une queue d'exploitation FIFO et les blocs mémoire sont alloués par algorithme de placement « first fit ».

Soit un cas de partage de la zone mémoire entre 5 processus A, B, C, D et E. On considère la séquence d'évènement suivante (a) à (f), ou les processus rentrent et quittent le système tour à tour. Les () indiquent pour chacun des évènements les besoins mémoires de chacun des processus. Donnez pour les évènements (a) à (f) l'organisation de la mémoire en précisant les ids des processus, les numéros de partition, les tailles des espaces mémoires occupés et vides. Indiquez finalement pour les évènements (e) (f) les taux de fragmentation mémoire.

(a)	(b)	(c)	(d)	(e)	(f)
A(96 Mo)	B(34 Mo)	C(18 Mo)	A(quitte)	D(48 Mo)	E(8Mo)

1.2.Partitionnement fixe à taille inégale

On considère une mémoire de 240 Mo gérée en partitionnement fixe en 4 partitions de taille inégale. Ces partitions sont de taille 2^{k+i} Mo ou i est le numéro de partition (0, 1, 2 et 3) et k=4 (i.e. 16, 32, 64 et 128 Mo).

Soit le cas de partage de la zone mémoire précédent de la question Q11 entre les 5 processus A, B, C, D et E. Au sein du système, l'allocation est traitée par une queue d'exploitation FIFO et les blocs mémoire sont alloués par algorithme de placement « best fit »¹. Donnez pour les évènements (a) à (f) l'organisation de la mémoire en précisant les ids des processus, les numéros de partition, les tailles des espaces mémoires occupés et vides. Indiquez finalement pour les évènements (e) (f) les taux de fragmentation mémoire.

Comparez avec le système de partitionnement fixe à taille égale, que pouvez-vous en conclure? Dans ce cas précis, compte tenu de la séquence d'allocation donnée que pouvez-vous dire de l'impact de l'algorithme « best fit ».

2. Gestion mémoire dynamique avec liste chaînée

On considère une mémoire de 200 Mo gérée par allocation dynamique en liste chaînée. Soit un cas de partage de cette mémoire entre 8 processus P1 à P8. A l'instant t₀ la mémoire est organisée comme décrite ci-dessous avec P1 à P8 les identifiants des processus et H (i.e.

¹ Ce qui est toujours le cas avec le partitionnement fixe à taille inégale.

« Hole ») les zones mémoire libres. Les éléments chiffrés dans la deuxième ligne indiquent pour chacun des processus les besoins mémoires en Mo.

P1	H	P4	P5	H	P2	H	P3	P6
32 Mo	16 Mo	64 Mo	8 Mo	4 Mo	18 Mo	22 Mo	6 Mo	30 Mo

On considère la séquence d'évènement suivante (a) à (d), où les processus rentrent et quittent le système tour à tour. Donnez pour chacun des évènements (a) à (d) l'organisation de la mémoire pour les algorithmes « first fit » et « next fit » en précisant les ids des processus, les tailles des espaces mémoires occupés et vides. Pour l'algorithme « next fit », on considérera comme dernier placement le processus P2(18 Mo).

(a)	(b)	(c)	(d)
P7(8 Mo)	P2(quitte)	P3(quitte)	P8(22 Mo)

Indiquez pour les deux algorithmes le taux de fragmentation mémoire à l'évènement (d), que pouvez-vous en conclure. Dans le cas de l'algorithme « next fit », expliquez ce qui se passe à l'évènement (d) et les contraintes sur la modélisation de la liste chaînée.

3. Algorithme du compagnon

On considère une mémoire de 256 Mo gérée par allocation hybride via l'algorithme du compagnon « buddy memory allocation ». Au sein de ce système, les blocs sont de taille 2^K Mo avec $4 \leq K \leq 8$. Soit un cas de partage de cette zone mémoire entre 10 processus A, B, C, D, E, F, G, H, I et J arrivant en mode FIFO (de A à J) dans le système. Les éléments chiffrés dans la deuxième ligne indiquent pour chacun des processus les besoins mémoires en Mo.

A	B	C	D	E	F	G	H	I	J
18 Mo	28 Mo	9 Mo	24 Mo	11 Mo	19 Mo	4 Mo	29 Mo	7 Mo	13 Mo

Indiquez pour la séquence considérée:

- le niveau de l'arborescence mise en œuvre au sein l'algorithme du compagnon.
- l'affectation de chacun des processus à un niveau i donné.
- le partitionnement mémoire à la fin du chargement FIFO des processus A à J, vous pourrez pour cela tracer l'arbre en fin de chargement de la séquence en indiquant les nœuds alloués et libres.
- pour chacun des évènements A à J, la constitution (taille) des listes (i.e. « buddies ») de taille 2^K Mo avec $4 \leq K \leq 8$.
- le nombre minimal et maximal de recherches récursives mises en œuvre et les processus associés.
- le taux de fragmentation mémoire en fin de séquence.

4. Pagination simple

Soit le cas d'une mémoire de 4 Go gérée par pagination simple. Au sein de ce système, la taille des pages est fixée à 64 Ko. Les unités Go et Ko sont à prendre ici au sens préfixe binaire (e.g. 1 Ko = 2^{10} octets). Considérant ces éléments, déterminez la structure d'adressage en pagination en précisant les codages (i.e. longueur en bits) de p , d et m . Considérant une TLB de 1024 entrées, quelle espace mémoire maximal peut-être représenté.

On considère maintenant un cas de partage de cette mémoire entre 5 processus A, B, C, D et E. Le tableau ci-dessous indique les événements (a) à (h) d'arrivée et de sortie des processus dans la mémoire avec pour chacun des processus les besoins mémoires en Kio (Kilo octet en préfixe binaire). Pour les besoins de l'exercice, on considère ici le cas de processus à faible demande mémoire. A l'instant t_0 , la mémoire est vide.

(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
A(140)	B(96)	C(220)	B(quitte)	D(205)	A(quitte)	C(quitte)	E(550)

Indiquez pour la séquence considérée:

- les besoins des processus A à E en nombre de page.
- le partitionnement de la mémoire pour les 8 événements (a) à (h), vous pourrez pour cela ne modéliser que la zone mémoire basse de 0 à 900 Ko.
- les tables de pages pour les processus actifs à l'évènement (h).
- considérant le cas de l'adresse logique 8F0F0 codée en hexadécimal pour le processus E, indiquez quelle adresse physique sera générée suite à l'évènement (h).
- le taux de fragmentation à l'évènement (e).