

Solution de Série d'exercices N°1 / chapitre : gestion de la mémoire centrale

Exercice N°1 :

-Taille de l'escape à allouer au processus = $32 - 8 = 24$ Mo.

- Pour chaque algorithme d'allocation, on représente l'état de la MC à chaque instant d'arrivée d'un processus ainsi que à chaque instant de sortie (fin de la durée d'exécution d'un processus).

- a) **First-Fit:** A chaque fois un processus demande d'un espace en MC, le SE va allouer la première partition dont la taille est suffisamment grande. La liste est ordonnée selon les adresses croissantes.

| Instant (t) | Partition système (8Mo) | Taille Usagers (24Mo) | | | |
|-------------|-------------------------|-----------------------|-----|------|-------|
| 0 | 8Mo | P1 | | 14Mo | |
| 4 | // | P1 | P2 | 8Mo | |
| 6 | // | 10Mo | | P2 | 8Mo |
| 7 | // | P3 | 8Mo | P2 | 8Mo |
| 10 | // | P3 | 8Mo | P2 | 8Mo |
| 14 | // | P3 | P4 | | 12 Mo |
| 15 | // | P3 | P4 | | 12 Mo |
| 16 | // | 2 Mo | P4 | | 12 Mo |
| 20 | // | P5 | | | P6 |
| 25 | // | 24Mo | | | |

- b) **Best-Fit:** A chaque fois un processus demande d'un espace en MC, le SE va Allouer la partition dont la taille est la plus proche de la taille demandée. La liste est ordonnée (ordre croissant) selon les tailles des partitions.

| Instant (t) | Partition système (8Mo) | Taille Usagers (24Mo) | | | |
|-------------|-------------------------|-----------------------|-----|------|----------|
| 0 | 8Mo | P1 | | 14Mo | |
| 4 | // | P1 | P2 | 8Mo | |
| 6 | // | 10Mo | | P2 | 8Mo |
| 7 | // | 10Mo | | P2 | P3 6Mo |
| 10 | | P4 | P2 | P3 | 6Mo |
| 14 | // | P4 | 6Mo | P3 | 6 Mo |
| 15 | // | P4 | 6Mo | P3 | 6 Mo |
| 16 | // | P5 | | | P6 |
| 21 | // | 24Mo | | | |

- c) **Worst-fit :** A chaque fois un processus demande d'un espace en MC, le SE va allouer la plus grande zone libre. La liste est ordonnée (ordre décroissant) selon les tailles des partitions.

Dans cette exercice **Worst-fit donne le même résultat comme First-fit.**

Exercice N°2 :

1.

| |
|-------------------|
| Programme User 1 |
| Debut |
| ... |
| @ :4000 procédure |
| Fin. |

| |
|------------------|
| Programme User 2 |
| Debut |
| @ :0 procédure |
| ... |
| Fin. |

| | |
|-----|------|
| 0 | |
| 1 | |
| 2 | |
| 3 | |
| 4 | 4000 |
| 5 | 4500 |
| 6 | 4501 |
| 7 | 6000 |
| ... | |

Table de pages User 1

| | |
|------|--------|
| | |
| | |
| 4000 | Page 0 |
| | |
| 4500 | Page1 |
| 4501 | Page2 |
| | |
| 6000 | Page 3 |
| | |

Mémoire centrale

| | |
|-----|------|
| 0 | 4000 |
| 1 | 4500 |
| 2 | 4501 |
| 3 | 6000 |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| ... | |

Table de pages User 2

2.

| | N° Octets référencé | (N°page | Déplacement) | (N°case | Déplacement) |
|--------------|---------------------|---------|--------------|---------|--------------|
| User1 | 0 | 4 | 0 | 4000 | 0 |
| | 2048 | 6 | 48 | 4501 | 48 |
| | 3100 | 7 | 100 | 6000 | 100 |
| | | | | | |
| User2 | 1000 | 1 | 0 | 4500 | 0 |
| | 2048 | 2 | 48 | 4501 | 48 |
| | 3500 | 3 | 500 | 6000 | 500 |

Exercice N°3 :

- Taille Table de pages (espace virtuel réellement utilisé):
 $\text{Nbr_Etrnées} = \text{Taille Espace Virtuel} / \text{Taille page} = 512\text{K} / 512 = 1024 \text{ entrées}$
 $\rightarrow \text{Taille Table Pages TailleT.P} = \text{Nbr_Etrnées} \times \text{Taille Entrée} = 4 \times 1024 = 4 \text{ Ko}$
- Taille maximale de la table de pages
 $\text{Nbr_Entrées} = \text{Taille Espace Virtuelle Max} / \text{Taille page} = 2^{24} / 512 = 2^{15}$
 $\rightarrow \text{Taille Table Pages TailleT.P} = \text{Nbr_Etrnées} \times \text{Taille Entrée} = 4 \times 2^{15} = 128 \text{ Ko}$
- Taille T.P = 128Ko, Espace réservé aux T.Pages = 6Mo X 5% = 0,3Mo
 $\text{Nbr TPages} = 0,3 \text{ Mo} / 128 \text{ Ko} = 2,4$. Nbr Tpages = 2.
 $\text{Nbr Programmes} = 10 \text{ Mo} / 512 \text{ Ko} = 20$.

Nbr utilisateurs = $\text{Min}(2, 20) = 2$

Exercice N°4 :

1. Chaîne de référence :

Calcul le nombre de pages dans l'espace virtuel :

$\text{Nb_Page} = \text{Taille espace Virtuelle} / \text{Taille de la page} = 8.$

$\text{N}^\circ \text{page} = \lfloor \text{adresse virtuelle} / \text{Taille de la page} \rfloor.$

Chaîne de référence : $\omega = 0, 5, 0, 1, 3, 6, 3, 2, 3, 0$

2. Algorithmes de remplacement (3 cases)

① FIFO (remplacer la page premièrement chargée en mémoire)

| | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|
| Chaîne de référence | → | 0 | 5 | 0 | 1 | 3 | 6 | 3 | 2 | 3 | 0 |
| Cases (pages) | } | 0 | 0 | 0 | 0 | 3 | 3 | 3 | 3 | 3 | 0 |
| | | | 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 |
| | | | | | 1 | 1 | 1 | 1 | 2 | 2 | 2 |
| Défauts de page | → | X | X | | X | X | X | | X | | X |

Nombre de défauts de page = 7.

② LRU (moins récemment utilisée)

| | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|
| Chaîne de référence | → | 0 | 5 | 0 | 1 | 3 | 6 | 3 | 2 | 3 | 0 |
| Cases (pages) | } | 0 | 0 | 0 | 0 | 0 | 6 | 6 | 6 | 6 | 0 |
| | | | 5 | 5 | 5 | 3 | 3 | 3 | 3 | 3 | 3 |
| | | | | | 1 | 1 | 1 | 1 | 2 | 2 | 2 |
| Défauts de page | → | X | X | | X | X | X | | X | | X |

Nombre de défauts de page = 7.

③ OPT

| | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|
| Chaîne de référence | → | 0 | 5 | 0 | 1 | 3 | 6 | 3 | 2 | 3 | 0 |
| Cases (pages) | } | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | 5 | 5 | 5 | 3 | 3 | 3 | 3 | 3 | 3 |
| | | | | | 1 | 1 | 6 | 6 | 2 | 2 | 2 |
| Défauts de page | → | X | X | | X | X | X | | X | | |

Nombre de défauts de page = 6.

Exercice N°5 :

1. Pour le processus 1 ($l = 0$), on a la chaîne de référence suivante : $\omega = 0, 1, 2, 3, 4, 1, 2, \dots$

| | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|
| Chaîne de référence | → | 0 | 1 | 2 | 3 | 4 | 0 | 1 | 2 | 3 | 4 |
| Cases (pages) | } | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | | | | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | | | | | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| | | | | | | 4 | 4 | 4 | 4 | 4 | 4 |
| Défauts de page | → | X | X | X | X | X | | | | | |

Nombre de défauts de pages = 5 par processus.

2. Introduction du processus N° 6 : chaque processus occupe 4 cases en mémoire centrale.

| Chaîne de référence | → | 0 | 1 | 2 | 3 | 4 | 0 | 1 | 2 | 3 | 4 | |
|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|
| Cases (pages) | } | | 0 | 0 | 0 | 0 | 4 | 4 | 4 | 4 | 3 | 3 |
| | | | | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 4 |
| | | | | | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| | | | | | | 3 | 3 | 3 | 3 | 2 | 2 | 2 |
| Défauts de page | → | X | X | X | X | X | X | X | X | X | X | |

Autant de défaut de page que de référence → Taus de défauts de page = 100%.