

Ordonnancement

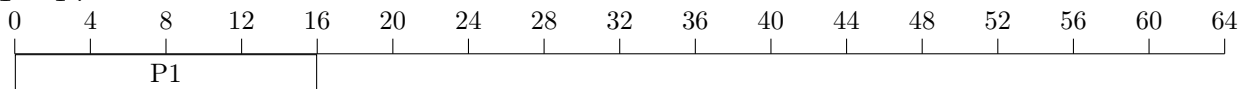
Valeran MAYTIE

Exercice 1 – Questions de cours

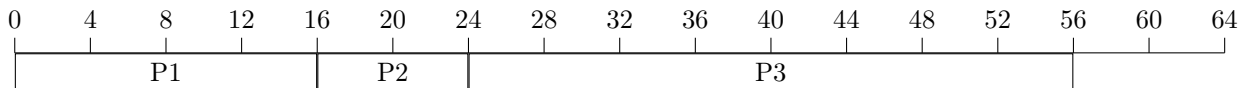
1. C'est le fait de passer d'une adresse symbolique à l'adresse mémoire
2. C'est dépend des cas ça se joue entre *Worth-Fit* et *Best-Fit*
3. On peut avoir des segment paratgés et on peut gérer les droits.
4. En comparant la limite et le décalage si décalage $>$ limite alors on a un débordement

Exercice 2 – Mémoire contiguë :

1. L'adresse physique doit faire 16 bit
2. Comme se n'est pas précisé l'adresse logique fait la même taille que l'adresse physique.
3. $T = 1$:

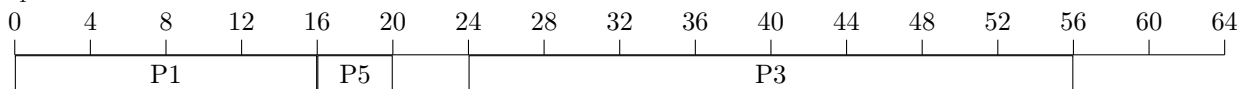


$T = 2$:

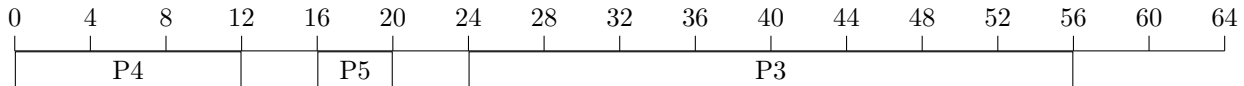


$T = 8$:

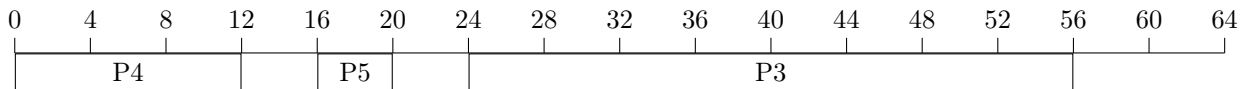
P_4 en attente



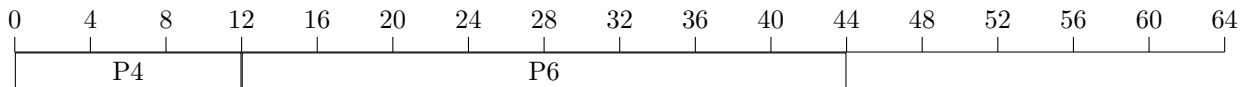
$T = 10$:



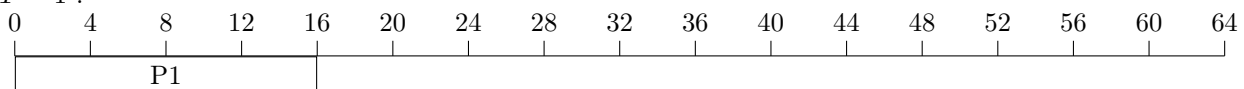
$T = 12$:



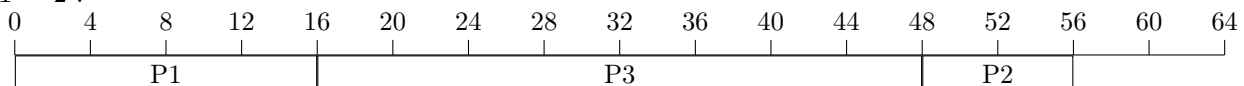
$T = 14$:

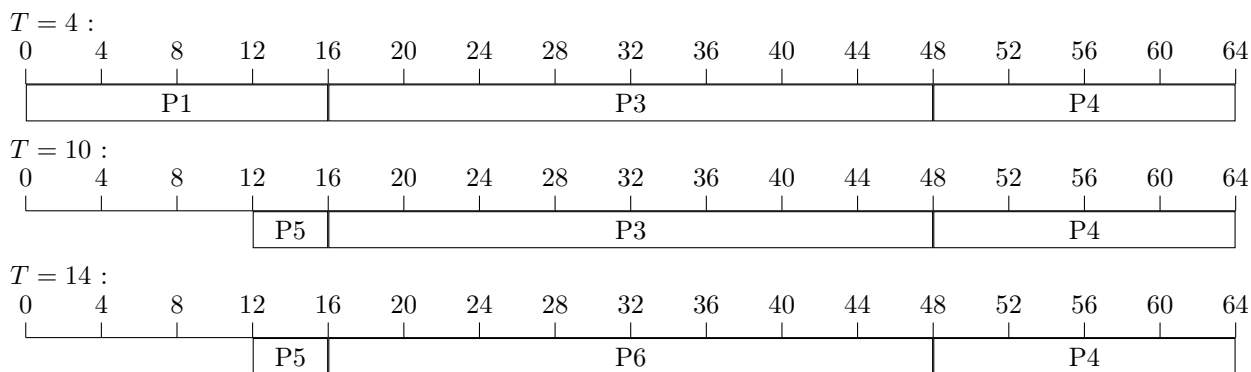


4. le dernier processus ce termine à $T = 22$
5. $T_8 : 2/52$
 $T_{10} : 4/48$
 $T_{12} : 4/48$ Mais P_6 en attente
6. Les deux autres méthode ne changent pas grand choses
7. $T = 1$:



$T = 2$:



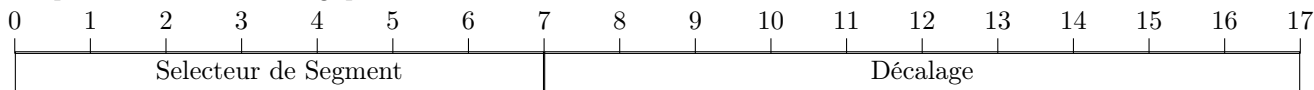


Exercice 3 – Segmentation simple

- $2^{16} = 65536$
- $2^{12} = 4096$
- Justification avec décalage $<$ limite
 $0x0430$: Segment 0 @physique = $0x0430 + 0x02B9 = 0x06E9$
 $0x1010$: Segment 1 @physique = $0x1010 + 0x2A00 = 0x2A10$
 $0x2B0F$: Segment 2 @physique = $0x2B0F + 0x0090 = 0x2B9F$
 $0x34FF$: Segment 3 @physique = $0x34FF + 0xC3A7 = 0xC8A6$
 $0x4100$: Segment 3 @physique = $0x4100 + 0x1D52 = 0x1E52$
- $0x5FF + 0x014 + 0x100 + 0x5D0 + 0x09F = 0xD82$
 Le processus à 3458 octets

Exercice 4 – Mini segmentation

- l'adresse physique fait 14 bits
- La taille maximale d'un segment est $2^4 + 2^6 = 2^{10} = 1024$
- Composition de l'adresse logique :



- Selecteur de segment sur 6 bits :
 $0xA8C7$: (Selecteur de segment : $0x2A$) (Décalage : $0x0C7$)
 On n'a pas $0x0C7 < (0x3 * 64 = 0xC0)$ donc Seg fault
 $0x1A2C$: (Selecteur de segment : $0x06$) (Décalage : $0x22C$)
 On n'a pas $0x22C < (0x4 * 64 = 0x100)$ donc Seg fault

Exercice 5 – Segmentation avancée

- Il y a 1Mo de mémoire physique, c'est à dir 2^{20} octets, donc l'adresse physique font 20bits.
- $(256 = 2^8)$ et $(64Ko = 2^{16})$ l'adresse logique fait donc $8 + 16 = 24$
- bits par ligne
12bits : base $(20 - 8)bits$ car multipli de 256
16bits : limite des 64Ko
4bits : Pour les droits
 On a donc $32bits = 4o$ par ligne donc la table fait $4 \times 128 = 512o$
- $0x2A04E5$: le bit de points fort est à 0 donc local à P_1
 Segment = $0x2A$, et $0x04E5 < 0xD704$
 L'accès en écriture est autorisé donc c'est possible d'écrire à cette adresse.
 — $0xA10386$: le bit de points fort est à 1 donc globale
 Segment $0x2A$, mais $0x0386 > 0x0100$
 Le décalage est trop grand il y aura une seg fault
 — $0x0c1F5E$: le bit de points fort est à 0 donc local à P_1
 Segment = $0x0C$, et $0x1F5E < 4000$ mais le bit W est égale à 0
 le décalage est correcte mais on n'a pas les permissions donc on ne peut pas écrire dedans

5. Il est probablement stocké dans 0x0C de la table locale voir dans le 0x6C dans la table global, mais il serai préférable de le mettre dans la table loale.
6. Ils ont la même position dans l'adresse physique mais avec des permissions différente. Cela permet à l'OS de paratger ça mémoire sans que le processus ne puis la modifier.
7. C'est un segment vide sans aucune permissions. Utilisé pour le pointeur NULL et repérer des erreurs de programtion.
8. L'OS peut réserver quelques segments dans la table globale pour partager son code et ses données. Les segments globaux sont accécible par tous le monde sans reconfigurer la MMU et avec le bit S il peut être rendu inaccécible pour des raisons de sécurité.

Lors d'une interruption l'OS n'a pas besoins de reconfigurer la MMU. Il n'a qu'a utilisé la configuration du processus courant.

9. Il est déjà possible pour un processus d'adresser une telle quantité de mémoire. Mais les adresses de base des segments ne couvrent pas plus de 1Mo de mémoire physique.

On ne peut pas augmenter la taille car sinon une ligne ferra plus de 32bits Il faudrait que les segments commencent sur des adresse physique mulitple de 1024, comme ça on pourra enlever les 1à bits de poinds faible.