



Module SETR

TD n° 2

Ordonnancement

Exercice 1 :

On considère un système constitué de trois tâches périodiques indépendantes :

Ta : (Ca=20, Ra=100, Ta=100)

Tb : (Cb=78, Rb=150, Tb=150)

Tc : (Cc=30, Rc=145, Tc=160)

1. On applique l'ordonnancement RM pour les trois tâches A, B et C. Calculer à l'aide du test de terminaison l'instant auquel la tâche Tc termine sa première exécution. Respecte-t-elle son échéance ?
2. On décide d'allouer les priorités aux tâches en fonction de leur échéance (analyse Deadline Monotonic). Utiliser le test suffisant de l'analyse RM pour prouver l'ordonnancabilité des tâches A et C. Pour ce faire, on considère que la tâche C a une capacité égale à $Cc + (Tc - Rc)$.
3. On ajoute la tâche B, utiliser le test de terminaison pour trouver à quel instant se termine la première exécution de B (pour pouvoir utiliser le test, on considère C comme si elle ajoutait un facteur de blocage Bb à la tâche B, égal à sa durée d'exécution (Cc).

Exercice 2 :

Les mêmes tâches A, B et C partagent maintenant une même ressource protégée par un sémaphore d'exclusion mutuelle. On suppose que l'algorithme utilisé par le système pour donner le droit d'accès à la ressource est de type "plafond de priorité". Sachant que le temps passé par chaque tâche en section critique est de 10 unités de temps, indiquer pour chacune des tâches A, B et C quel est le temps de blocage maximal d'inversion de priorité.

Vérifier que les trois tâches sont toujours ordonnancables en tenant compte de ces nouveaux temps de blocage.

Exercice 3 :

Un programme consiste en cinq (5) tâches A, B, C, D et E (en ordre décroissant de priorité, A étant la plus prioritaire) et six (6) ressources R1 à R6 qui sont protégées par des sémaphores réalisant le protocole « priority ceiling ». Le temps d'accès maximal aux ressources est donné par :

| | | | | | |
|-------|--------|-------|--------|--------|--------|
| R1 | R2 | R3 | R4 | R5 | R6 |
| 50 ms | 150 ms | 75 ms | 300 ms | 250 ms | 175 ms |

Les ressources sont utilisées par les tâches de la façon suivante :

| Tâche | Ressource |
|-------|-----------|
| A | R3 |
| B | R1,R2 |
| C | R3,R4,R5 |
| D | R1,R5,R6 |
| E | R2,R6 |

Calculer le blocage maximal pour chacune des tâches.