

## Semaine 12 – Ordonnancement temps réel – Solutions

1. Qu'est-ce qu'un système d'exploitation temps réel? Identifiez les principales caractéristiques. Nommez deux systèmes d'exploitation temps réel dédiés (ou durs).

Réponse :

Assure les fonctions de base d'un système d'exploitation (gestion des tâches, gestion des interruptions, gestion du temps, gestion de la mémoire, etc.)

En plus :

- garantir des temps de réponses bornés et acceptables aux demandes de services,
- permettre de développer et de mettre en œuvre des applications temps réel (tâches concurrentes, communicantes avec éventuellement des contraintes temporelles)

QNX, MicroC, VxWorks

2. Nommez deux types d'ordonnancement de tâches temps réel.

1. Ordonnancement de tâches indépendantes

- a. Algorithme à priorités fixes avec affectation des priorités Rate Monotonic priority assignment (RMA) : Ordonnancement préemptif à priorités statiques où la priorité d'une tâche est inversement proportionnelle à sa période
- b. Algorithme à priorités dynamiques Earliest Deadline First (EDF) : Ordonnancement préemptif à priorités statiques où la priorité d'une tâche est inversement proportionnelle à son échéance

2. Ordonnancement de tâches périodiques dépendantes

- a. Inversion des priorités : **Protocole d'héritage de priorités (PIP)**, OCPP, ICPP

3. Comment détermine-t-on l'ordonnançabilité des tâches avec RMA? Expliquez.

Réponse :

Un ensemble de  $n$  tâches périodiques indépendantes telles que  $D_i = P_i$  pour  $i=1$  à  $n$ , est ordonnançable si (condition **suffisante** de Liu et Layland) :

$$\sum_{i=1}^n \frac{C_i}{P_i} \leq n \left( 2^{1/n} - 1 \right)$$

### Ordonnancement de tâches indépendantes

4. Une application logicielle est composée de 2 tâches : T1 et T2.

Les caractéristiques de ces tâches sont les suivantes :

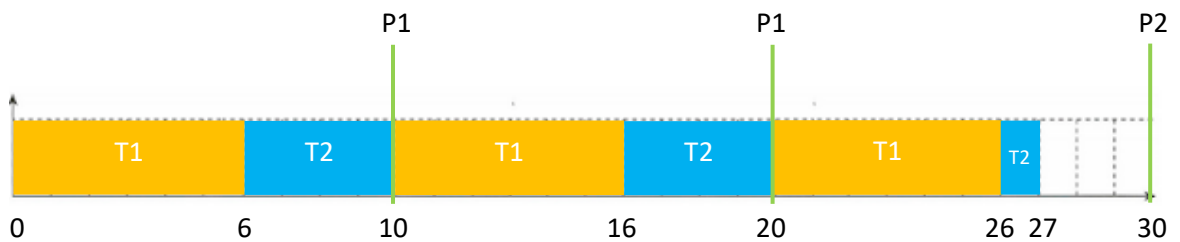
T1 :  $C1 = 6$ ,  $P1 = 10$

T2 :  $C2 = 9$ ,  $P2 = 30$

Élection : la priorité la plus grande d'abord.

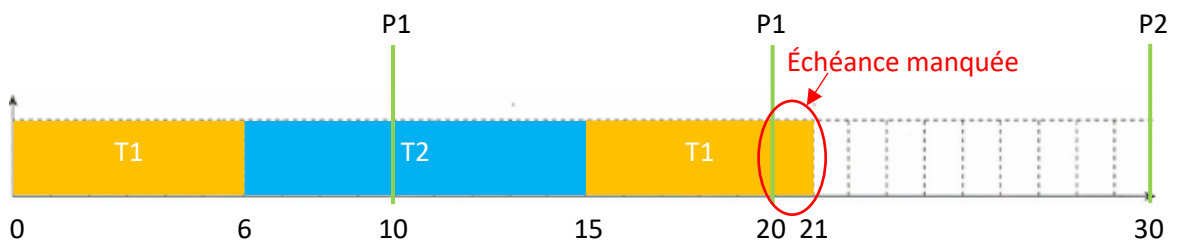
- a) Dessinez le diagramme de Gantt si on utilise l'algorithme RMA préemptif

Réponse : Ordonnançable



- b) Dessinez le diagramme de Gantt si on utilise l'algorithme RMA non préemptif

Réponse : Non ordonnançable



5. Une application logicielle est composée de 2 tâches : T1 et T2.

Les caractéristiques de ces tâches sont les suivantes :

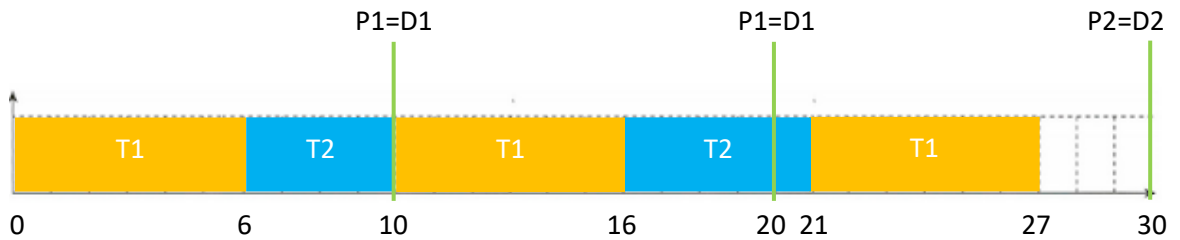
T1 :  $C1 = 6$ ,  $P1 = D1 = 10$

T2 :  $C2 = 9$ ,  $P2 = D2 = 30$

Élection : la plus courte échéance d'abord.

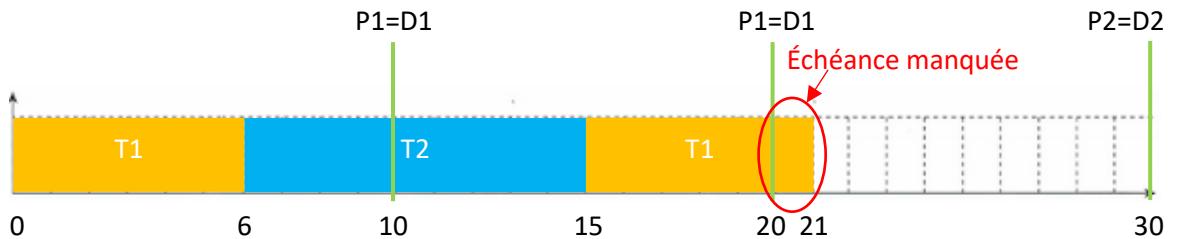
a) Dessinez le diagramme de Gantt si on utilise l'algorithme EDF préemptif

Réponse : Ordonnançable



b) Dessinez le diagramme de Gantt si on utilise l'algorithme EDF non préemptif

Réponse : Non ordonnançable



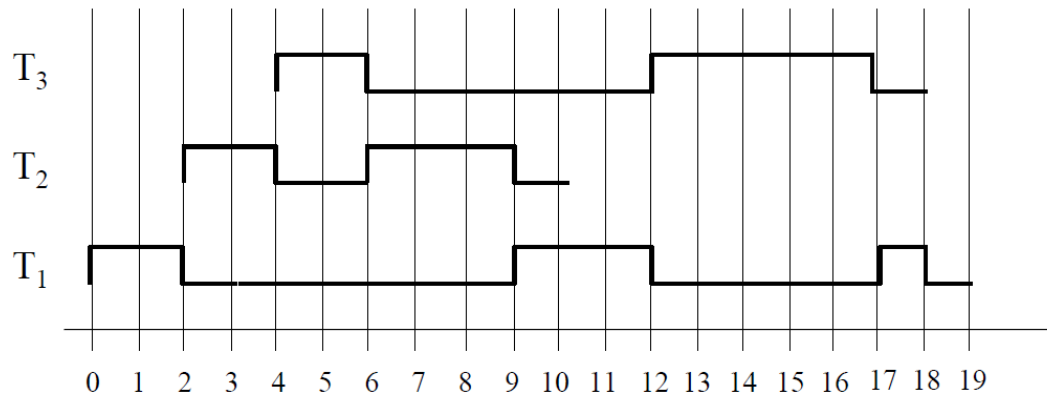
### Ordonnement de tâches périodiques dépendantes

6. Une application logicielle est composée de 3 tâches : T1, T2 et T3. Les caractéristiques de ces tâches sont données dans le tableau ci-dessous. La tâche T3 est la tâche la plus prioritaire et la tâche T1 la moins prioritaire. Dans la colonne Exécution la lettre E indique l'exécution d'une tâche pendant une période (un *tick*), la lettre A indique l'entrée dans la section critique A pendant une période et la lettre B indique l'entrée dans la section critique B pendant une période.

Tâche	Priorité	Date d'arrivée	Exécution
T1	1	0	EAAAAE
T2	2	2	EBBBE
T3	3	4	EEAABEE

- a) Dessinez le diagramme de Gantt si on utilise un ordonnancement préemptif à priorités.

Réponse :



- b) Est-ce que votre ordonnancement illustre correctement l'ordre d'exécution des trois tâches selon les priorités assignées? Justifiez.

Réponse :

Oui, le diagramme de Gantt illustre correctement l'ordre d'exécution des trois tâches selon les priorités assignées. Cependant, il y a un problème d'inversion des priorités. La tâche T3 (la plus prioritaire) est bloquée par une tâche moins prioritaire (T1) et son exécution finie après l'exécution d'une tâche moins prioritaire (T2).

- c) Y a-t-il un problème d'inversion des priorités? Si oui, donnez le diagramme de Gantt en utilisant le protocole PIP (Priority Inheritance Protocol) et expliquez que votre nouveau diagramme corrige ce problème. Si non, expliquez clairement qu'il n'y a pas de problème d'inversion des priorités.

Réponse :

Oui, il y a un problème d'inversion des priorités. En utilisant le protocole PIP (voir diagramme ci-dessous) la tâche T1 devient plus prioritaire, elle hérite de la priorité de T3 sur l'utilisation de la ressource A, elle libère donc plus rapidement cette ressource. Ainsi, T3 se termine en premier, suivi de T2 et finalement T1 (respectant les priorités assignées).

