

**Exercice 01 : (Sur 10.00 points) Questions du cours**

**Question 1.1 : (Sur 05.00 points)**

1. Quel est le rôle de l'ordonnanceur temps réel ? (Sur 02.00 points).
2. Décrire brièvement l'ordonnanceur du système temps réel ? (Sur 02.00 points).
3. Qu'est-ce la commutation de contexte ? Expliquer. (Sur 01.00 point).

**Question 1.2 : (Sur 05.00 points)**

- Quelles sont les cinq (05) critères les plus souvent proposés pour comparer et évaluer les performances des algorithmes d'ordonnement temps réel du système d'exploitation temps réel ? Détailler votre réponse. (01.00 point par critère).

**Exercice 02 : (Sur 10.00 points) Application pratique de l'algorithme temps réel.**

Soit un ensemble de trois (03) tâches périodiques définies par leurs paramètres  $C_i$ ,  $D_i$  et  $P_i$  (donnés dans le tableau suivant) :

Tâches	Arrivé ( $r_i$ )	Temps d'exécution ( $C_i$ )	Délai critique ( $D_i$ )	Période ( $P_i$ )
$T_1$	0	2	5	6
$T_2$	0	2	4	8
$T_3$	0	4	8	12

**Question n°2.1 : (Sur 01.50 points)**

Que veut dire DMA ? Donner la définition de l'algorithme DMA.

**Question n°2.2 : (Sur 01.50 points)**

L'algorithme DMA est basé sur quoi ?

**Question n°2.3 : (Sur 02.50 points)**

Les tâches  $T_1$ ,  $T_2$  et  $T_3$  sont-elles ordonnançables par DMA ?

**Question n°2.4 : (Sur 04.50 points)**

Donner une séquence DMA valide en traçant le graphique d'exécution des tâches. Que conclure ?

Bon Courage

## **Solution du contrôle**

### **Solution de l'exercice 01 : (Sur 10.00 points) Questions du cours**

#### **Question 1.1 : (Sur 05.00 points)**

1. Quel est le rôle de l'ordonnanceur temps réel ? (Sur 02.00 points)
2. Décrire brièvement l'ordonnanceur du système temps réel ? (Sur 02.00 points)
3. Qu'est-ce la commutation de contexte ? Expliquer. (Sur 01.00 point)

#### **Réponse 1.1 : (Sur 05.00 points)**

1. L'ordonnanceur temps réel son rôle est de gérer l'allocation du processeur aux différents processus temps réel et processus normaux. Il gère aussi l'enchaînement et la concurrence des exécutions des processus, en optimisant l'occupation des unités de traitement selon des contraintes temporelles de chaque processus. C'est la partie logicielle d'une application temps réel, qui est un ensemble de processus synchronisés, communicants et partageant des ressources critiques. (Sur 02.00 points).
2. L'ordonnanceur temps réel, c'est le processus système qui gère l'ordonnancement des processus temps réel via un algorithme d'ordonnancement, qui est une méthode ou une stratégie utilisée pour ordonnancer les processus temps réel, il s'appuie sur la connaissance de certaines caractéristiques des processus statiques ou dynamiques : qu'ils soient périodiques ou aperiodiques, cycliques ou non cycliques, préemptifs ou non, à échéance et pire temps d'exécution des processus, et enfin à priorité fixe ou à échéance. (Sur 02.00 points).
3. Le fait de commuter le processeur sur un autre processus demande de sauvegarder l'état de l'ancien processus et de charger l'état sauvegardé par le nouveau processus. Cette tâche est connue sous le nom de commutation de contexte. (Sur 01.00 point).

#### **Question 1.2 : (Sur 05.00 points)**

- Quelles sont les cinq (05) critères les plus souvent proposés pour comparer et évaluer les performances des algorithmes d'ordonnancement temps réel du système d'exploitation temps réel ? Détailler votre réponse. (01.00 point par critère).

#### **Réponse 1.2 : (Sur 05.00 points)**

Les critères les plus souvent utilisés sont :

1. Utilisation du processeur : Un bon algorithme d'ordonnancement sera celui qui maintiendra le processeur aussi occupé que possible.
2. Capacité de traitement : C'est la quantité de processus terminés par unité de temps.
3. Temps de restitution : C'est le temps s'écoulant entre la soumission du travail et sa terminaison.
4. Temps d'attente : C'est le temps passé à attendre dans la file d'attente des processus prêts.
5. Temps de réponse : C'est le temps passé dans la file d'attente des processus prêts avant la première exécution.

**Exercice 02 : (Sur 10.00 points) Application pratique de l'algorithme temps réel.**

Soit un ensemble de trois (03) tâches périodiques définies par leurs paramètres  $C_i$ ,  $D_i$  et  $P_i$  (donnés dans le tableau suivant) :

Tâches	Arrivé ( $r_i$ )	Temps d'exécution ( $C_i$ )	Délai critique ( $D_i$ )	Période ( $P_i$ )
$T_1$	0	2	5	6
$T_2$	0	2	4	8
$T_3$	0	4	8	12

**Question n°2.1 : (Sur 01.50 points)**

Que veut dire DMA ? Donner la définition de l'algorithme DMA.

**Question n°2.2 : (Sur 01.50 points)**

L'algorithme DMA est basé sur quoi ?

**Question n°2.3 : (Sur 02.50 points)**

Les tâches  $T_1$ ,  $T_2$  et  $T_3$  sont-elles ordonnançables par DMA ?

**Question n°2.4 : (Sur 04.50 points)**

Donner une séquence DMA valide en traçant le graphique d'exécution des tâches. Que conclure ?

**Solution de l'exercice 02 : (Sur 10.00 points) Application pratique de l'algorithme temps réel.**

Soit un ensemble de trois (03) tâches périodiques définies par leurs paramètres  $C_i$ ,  $P_i$  et  $D_i$  (donnés dans le tableau suivant) :

Tâches	Arrivé ( $r_i$ )	Temps d'exécution ( $C_i$ )	Délai critique ( $D_i$ )	Période ( $P_i$ )
T <sub>1</sub>	0	2	5	6
T <sub>2</sub>	0	2	4	8
T <sub>3</sub>	0	4	8	12

**Question n°2.1 : (Sur 01.50 points)**

Que veut dire DMA ? Donner la définition de l'algorithme DMA.

**Réponse 2.1 : (Sur 01.50 points)**

DMA veut dire Deadline Monotonic Analysis.

C'est un algorithme à priorité constante pour processus cycliques (le processus le plus prioritaire est celui de plus petite échéance). C'est-à-dire que la priorité  $T_2 >$  à la priorité  $T_1 >$  à la priorité  $T_3$ .

**Question n°2.2 : (Sur 01.50 points)**

L'algorithme DMA est basé sur quoi ?

**Réponse n°2.2 : (Sur 01.50 points)**

L'algorithme DMA est basé sur le délai critique. La tâche de plus petit délai critique est la plus prioritaire.

**Question n°2.3 : (Sur 02.50 points)**

Les tâches T<sub>1</sub>, T<sub>2</sub> et T<sub>3</sub> sont-elles ordonnançables par DMA ?

**Réponse n°2.3 : (Sur 02.50 points)**

Pour pouvoir voir, si les tâches sont ordonnançables ou non par DMA, il va falloir effectuer le test d'acceptabilité. C'est-à-dire vérifier la condition d'acceptabilité suivante :

La condition d'acceptabilité : (Sur 0.25 point)  $\sum_{i=0}^3 \frac{C_i}{D_i} \leq n (2^{1/n} - 1)$

Le tableau résume les calculs :

Tâches	$r_i$	$C_i$	$D_i$	$P_i$	$C_i/D_i$
$T_1$	0	2	5	6	0,4
$T_2$	0	2	4	8	0,5
$T_3$	0	4	8	12	0,5
Somme					1,40

(Sur 0.75 point)

Ainsi, on a :

Le calcul de la somme des  $C_i$  sur  $D_i$  : (Sur 0.25 point)  $\sum_{i=0}^3 \frac{C_i}{D_i} = 1.40$

Le calcul de la quantité  $n (2^{1/n} - 1)$  pour  $n = 3$  : (Sur 0.25 point)

$$n (2^{1/n} - 1) = 3 (2^{1/3} - 1) = 0.7798 = 0.78$$

On constate que :

$$\left(\sum_{i=0}^3 \frac{C_i}{D_i} = 1.40\right) > (n (2^{1/n} - 1) = 0.78) \text{ (Sur 0.25 point)}$$

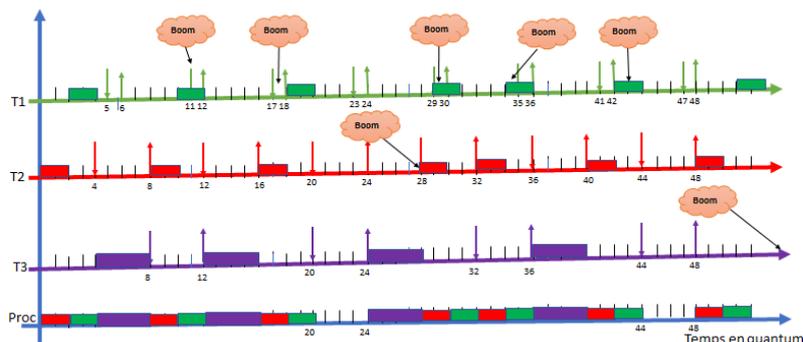
Dans ce cas, on ne pourrait pas ordonner les trois tâches  $T_1$ ,  $T_2$  et  $T_3$  sous DMA avec des risques de ne pas respecter les échéances lors des différents cycles d'exécution des trois tâches à cause de la condition d'acceptabilité. (Sur 0.75 point).

### **Question n°2.4 : (Sur 04.50 points)**

Donner une séquence DMA valide en traçant le graphique d'exécution des tâches. Que conclure ?

### **Réponse n°2.4 : (Sur 04.50 points)**

Dans ce cas, on va procéder à l'ordinogramme d'exécution sous DMA des trois tâches : (Sur 01.50 points)



**Contrôle du module : Système Temps Réel & Informatique Industrielle**

**Durée : 01H00**

**Date : 22 Mai 2022**

**Documents non autorisés**

On constate que l'exécution des trois tâches  $T_1$ ,  $T_2$  et  $T_3$  se déroule de façon normale durant le premier cycle d'exécution et c'est à partir du deuxième cycle que l'on commence à avoir des dépassements de délais de la tâche  $T_1$ , ensuite à partir du quatrième cycle celle de  $T_2$  et par la suite à partir du quatrième cycle celle de  $T_3$ , etc. (Sur 01.50 points).

On remarque la gestion des tâches par l'algorithme DMA est lourde et le processeur un moment donné est sans activité entre le 20 et le 24 quantum et ensuite entre le 44 et le 48 quantum. Peut-être, la solution réside pour améliorer l'algorithme est de ne pas considérer les priorités avec préemption et peut être l'algorithme RMA est meilleur dans ce cas, mais avec plus d'oisiveté du processeur ou l'algorithme Rand Robin ou le tourniquet peuvent gérer efficacement ces trois tâches. (Sur 01.50 points).