



**Exercice n°1 : (sur 14.00 points) Questions de cours**

**Question n°1.1 : (sur 11.00 points)** Quelles sont les sept (07) politiques d'ordonnancement temps réel des processus périodiques? Déterminez votre réponse selon le tableau suivant : (à refaire le tableau dans votre copie d'examen)

N°	Politiques d'ordonnancement des processus périodiques (préciser le sigle en anglais et sa signification en anglais)	Explications détaillées du principe de l'ordonnancement des processus périodiques
1	(Sur 0.50 point)	(sur 0.50 point)
2	(Sur 0.50 point)	(sur 0.50 point)
3	(Sur 0.50 point)	(sur 0.50 point)
4	(Sur 0.50 point)	(sur 01.50 points)
5	(Sur 0.50 point)	(sur 01.50 points)
6	(Sur 0.50 point)	(sur 01.50 points)
7	(Sur 0.50 point)	(sur 01.50 points)

**Question n°1.2 : (sur 03.00 points)** Quelles sont les trois (03) politiques d'ordonnancement temps réel des processus aperiodiques? Déterminez votre réponse selon le tableau suivant : (à refaire le tableau dans votre copie d'examen)

N°	Politiques d'ordonnancement des processus aperiodiques (préciser le sigle en anglais et sa signification en anglais si possible)	Explications détaillées du principe de l'ordonnancement des processus aperiodiques
1	(Sur 0.25 point)	(sur 0.75 point)
2	(Sur 0.25 point)	(sur 0.75 point)
3	(Sur 0.25 point)	(sur 0.75 point)

**Exercice n°2 : (sur 06.00 points)** Considérant l'ordonnancement basé sur RMA.

**Question n°2.1 : (sur 01.50 points)** Expliquez ce qu'on entend, lorsqu'on dit que la condition d'ordonnancement de Liu et Layland est une condition suffisante, mais pas nécessaire.

$$\text{Condition de Liu et Layland : } \sum_{i=1}^N \left( \frac{C_i}{T_i} \right) \leq N(2^{1/N} - 1)$$

**Question n°2.2 : (sur 04.50 points)** En utilisant la condition de Liu et Layland et l'ordonnancement graphique, que peut-on dire des tâches du tableau suivant du point de vue ordonnancement? Tracer le graphique d'exécution des tâches. Que conclure ?

Tâches	Période	Temps d'exécution
T <sub>1</sub>	50	10
T <sub>2</sub>	40	10
T <sub>3</sub>	30	15

Bonne chance



**Réponse n° 1.1** : (sur 05.00 points) Quelles sont les sept (07) politiques d'ordonnement des temps réel des processus périodiques ? Décrivez votre réponse selon le tableau suivant :

N°	<i>Politiques d'ordonnement des processus (préciser le sigle en anglais et sa signification en anglais)</i>	<i>Explications détaillées du principe de l'ordonnement des processus</i>
1	Premier arrivé, premier servi FIFO : First In First Out (sur 0.50 point)	<ul style="list-style-type: none"> <li>Si plusieurs processus ont le même temps d'exécution, la politique FIFO est alors utilisée pour les départager. (sur 0.50 point)</li> </ul>
2	Plus court d'abord SJF : Shortest Job First (sur 0.50 point)	<ul style="list-style-type: none"> <li>Affecte le processeur au processus possédant le temps d'exécution le plus court (sur 0.50 point)</li> </ul>
3	Par tourniquet Round robin (sur 0.50 point)	<p>Il est conçu pour les systèmes à temps partagé. Il alloue le processeur aux processus à tour de rôle pendant une période de temps appelée quantum de 10 à 100 ms. (sur 0.50 point)</p>
4	Par priorité fixe ou constante RMA : Rate Monotonic Analysis (sur 0.50 point)	<ul style="list-style-type: none"> <li>C'est un ordonnancement préemptif à priorité fixe ou constante. L'ordonnement à taux monotone est un algorithme d'ordonnement temps réel en ligne à priorité constante (statique).</li> <li>Il est basé sur la période (processus à échéance sur requête).</li> <li>Il attribue la priorité la plus forte à la tâche qui possède la plus petite période.</li> <li>RMA est optimal dans le cadre d'un système de tâches périodiques, synchrones, indépendantes et à échéance sur requête avec un ordonnanceur préemptif. De ce fait, il n'est généralement utilisé que pour ordonner des tâches vérifiant ces propriétés.</li> <li>L'activation en début de période (<math>S_i = 0</math>)</li> <li>L'échéance (<math>D_i</math>) en fin de période (<math>P_i</math>) avec (<math>D_i = T_i</math>)</li> <li>Pire cas <math>C_i</math> connu a priori.</li> <li>Choix du processus éligible de plus courte période.</li> <li>Test d'acceptabilité (condition suffisante). (sur 01.50 points)</li> </ul>
5	Par priorité fixe ou constante DMA : Deadline Monotonic Analysis (sur 0.50 point)	<ul style="list-style-type: none"> <li>C'est un ordonnancement préemptif à priorité fixe ou constante.</li> <li>Le principe est proche de RTM.</li> <li>Il est basé sur le délai critique.</li> <li>L'échéance (<math>D_i</math>) est inférieure à la période (<math>P_i</math>).</li> <li>Le classement des processus par échéances</li> </ul>





		<p>croissantes.</p> <ul style="list-style-type: none"> <li>• Si <math>T_i = D_i</math> devient un cas particulier de RMS.</li> <li>• La condition nécessaire et suffisante existe.</li> <li>• Test d'acceptabilité (condition suffisante).</li> </ul> <p>(sur 01.50 points)</p>
6	<p>Par priorité dynamique                  EDF : Earliest Deadline First                  (sur 0.50 point)</p>	<ul style="list-style-type: none"> <li>• C'est un ordonnancement préemptif à priorité dynamique ou variable.</li> <li>• Algorithme à priorité dynamique pour processus cycliques.</li> <li>• Il est basé sur l'échéance.</li> <li>• A chaque réveil de processus, la priorité maximale est donnée au processus dont l'échéance est la plus proche.</li> <li>• Test d'acceptabilité (condition nécessaire et condition suffisante).</li> </ul> <p>(sur 01.50 points)</p>
7	<p>Par priorité dynamique                  LLF : Least Laxity First                  (sur 0.50 point)</p>	<ul style="list-style-type: none"> <li>• C'est un ordonnancement préemptif à priorité dynamique ou variable.</li> <li>• Il est basé sur la laxité résiduelle.</li> <li>• La priorité maximale est donnée au processus qui a la plus petite laxité résiduelle - <math>L(t) = D(t) - C(t)</math>. Il est équivalent à EDF si on ne calcule la laxité qu'au réveil des processus.</li> </ul> <p>(sur 01.50 points)</p>

**Réponse n°1.2 :** (sur 03.00 points) Quelles sont les trois (03) politiques d'ordonnancement temps réel des processus aperiodiques? Décrivez votre réponse selon le tableau suivant : (à refaire le tableau dans votre copie d'examen)

N°	Politiques d'ordonnancement des processus aperiodiques (préciser le sigle en anglais et sa signification en anglais) si possible	Explications détaillées du principe de l'ordonnancement des processus aperiodiques
1	<p>Processus aperiodique par serveur de scrutation (polling)                  (un serveur se comporte comme un processus périodique)                  (sur 0.25 point)</p>	<ul style="list-style-type: none"> <li>• Une fois ce serveur lancé, s'il existe des tâches aperiodiques en attente, elles sont exécutées dans la limite de la période du serveur. Le cas échéant, le serveur se suspend, jusqu'à sa prochaine période d'occurrence.</li> <li>• Cette technique garantit une certaine périodicité dans le traitement des tâches aperiodiques.</li> <li>• En revanche, le rendement est faible et ceci est dû au fait que le serveur perd sa capacité de service quand il se suspend.</li> <li>• A chaque activation, traitement des processus en suspens jusqu'à épuisement de la capacité ou jusqu'à ce qu'il n'y ait plus de processus en attente.</li> <li>• Si aucun processus n'est en attente (à l'activation ou parce que le dernier processus a été traité).</li> </ul>



		<ul style="list-style-type: none"><li>Le serveur se suspend immédiatement et perd sa capacité qui peut être réutilisée par les tâches périodiques (amélioration du temps de réponse).</li><li>La capacité du serveur est réinitialisée périodiquement.</li></ul> (sur 0.75 point)
2	Processus aperiodique par serveur différé ou continu (un serveur se comporte comme un processus périodique) (sur 0.25 point)	<ul style="list-style-type: none"><li>Ce serveur termine sa capacité de service (son temps d'exécution) même s'il n'y a aucun processus à servir.</li><li>Ainsi moins de processus sont différés, en effet si un processus aperiodique arrive et que le serveur est en cours d'exécution, il est servi jusqu'à la fin de son temps d'exécution ou jusqu'à la fin du temps alloué au serveur.</li><li>L'inconvénient d'un pareil serveur est qu'il peut s'exécuter et ne trouver aucun processus aperiodique à tester, auquel cas son temps de service est perdu.</li><li>De plus la capacité du serveur est une donnée difficile à déterminer et peut être insuffisante face à de nombreuses créations de processus aperiodiques.</li><li>La capacité du serveur est réinitialisée périodiquement.</li></ul> (sur 0.75 point)
3	Processus aperiodique par serveur sporadique (un serveur se comporte comme un processus périodique) (sur 0.25 point)	<ul style="list-style-type: none"><li>Il améliore le temps de réponse des tâches aperiodiques sans diminuer le taux d'utilisation du processeur pour les tâches périodiques.</li><li>Comme le serveur ajournable mais ne retrouve pas sa capacité à période fixe.</li><li>Le serveur sporadique peut être considéré comme une tâche périodique "normale" du point de vue des critères d'ordonnancement.</li><li>Les processus aperiodiques dont le taux d'arrivée est borné supérieurement sont dits sporadiques.</li><li>La charge due aux processus aperiodiques est borné.</li><li>Il est adapté aux processus aperiodiques dont le temps de latence est faible.</li></ul> (sur 0.75 point)

**Exercice n°2 : (sur 06.00 points) Application de RMA**

**Réponse n°2.1 : (sur 01.50 points)**

La condition Liu et Layland est suffisante, mais pas nécessaire : si elle est satisfaite, les tâches peuvent être ordonnancées, sinon, on ne peut rien conclure.

**Réponse n°2.2 : (sur 04.50 points)**

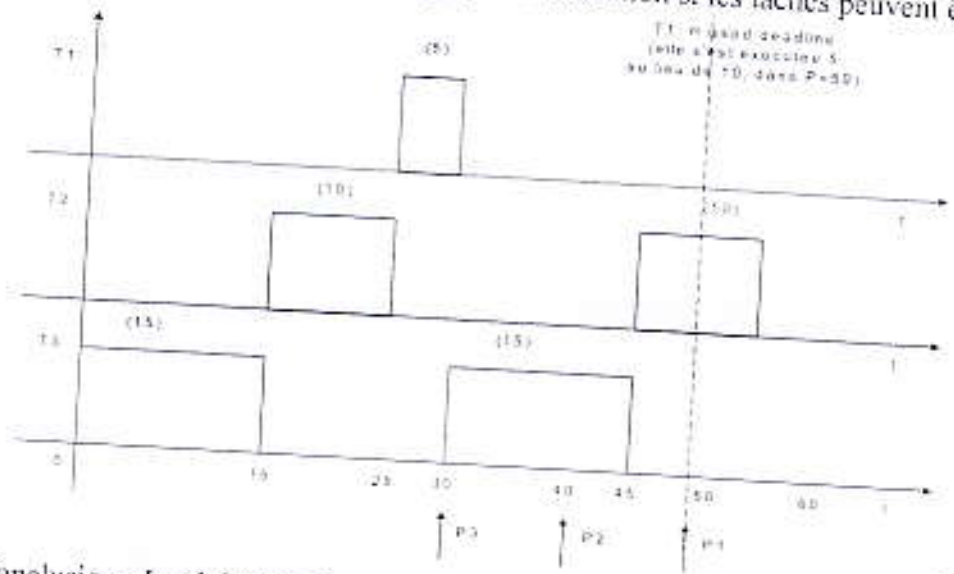
La condition de Liu et Layland n'est pas satisfaite : (sur 01.50 points)





$$(10/50) + (10/40) + (15/30) = 0.95 > 0.78$$

Alors il faut vérifier sur le graphe d'exécution si les tâches peuvent être ordonnancées :



Conclusion : La tâche T1 dépasse son deadline lors de sa première période, donc les tâches ne peuvent pas être ordonnancées par RMA. (sur 0.50 point)

(sur 02.50 points)