

ENSEEIH - 3SN

Contrôle de systèmes temps réel - 6 janvier 2023

Durée : 1 heure - Tous documents autorisés

Ordonnancement temps réel

Exercice 1

1. Soit la configuration de tâches périodiques indépendantes suivante :

- $T_1 : C = 2, D = 4, P = 4$
- $T_2 : C = 2, D = 6, P = 6$
- $T_3 : C = 1, D = 12, P = 12$
- $T_4 : C = 1, D = 24, P = 24$

Est-elle ordonnançable avec un algorithme à priorités statiques ? Si ce n'est pas le cas, l'est-elle avec un algorithme à priorités dynamiques ?

2. Même question pour la configuration de tâches périodiques indépendantes suivante :

- $T_1 : C = 2, D = 4, P = 4$
- $T_2 : C = 2, D = 2, P = 6$
- $T_3 : C = 1, D = 12, P = 12$
- $T_4 : C = 1, D = 11, P = 24$

3. Même question pour la configuration de tâches périodiques indépendantes suivante :

- $T_1 : C = 2, D = 4, P = 4$
- $T_2 : C = 2, D = 2, P = 6$
- $T_3 : C = 1, D = 10, P = 12$
- $T_4 : C = 1, D = 11, P = 24$

Exercice 2

Soit la configuration de tâches périodiques indépendantes suivante :

- $T_1 : C = 5, D = 10, P = 10$
- $T_2 : C = 5, D = 20, P = 20$

On y ajoute les deux requêtes apériodiques suivantes :

- $T_3 : r = 1, C = 2$
- $T_4 : r = 11, C = 1$

Quels sont les temps de réponse de ces tâches apériodiques

1. lorsqu'on les exécute en arrière-plan,
2. lorsqu'on utilise un serveur de scrutation de période 4 et de budget 1,
3. lorsqu'on utilise un serveur de ajournable de période 4 et de budget 1,
4. lorsqu'on utilise un serveur de sporadique De période 4 et de budget 1.

Exercice 3

Soit la configuration de tâches périodiques suivante, qui partagent les ressources R_1 , R_2 et R_3 :

	r_0	WCET	D	P						
T_1	10	3 : <table><tr><td></td><td></td><td></td></tr></table>				3	18			
T_2	4	3 : <table><tr><td></td><td>R_1</td><td></td></tr></table>		R_1		7	18			
	R_1									
T_3	2	5 : <table><tr><td></td><td>R_2</td><td>R_2</td><td>R_2R_3</td><td></td></tr></table>		R_2	R_2	R_2R_3		15	18	
	R_2	R_2	R_2R_3							
T_4	0	6 : <table><tr><td></td><td>R_1</td><td>R_1</td><td>R_1R_3</td><td>$R_1R_2R_3$</td><td></td></tr></table>		R_1	R_1	R_1R_3	$R_1R_2R_3$		18	18
	R_1	R_1	R_1R_3	$R_1R_2R_3$						

Cette configuration est-elle ordonnançable ?

Exercice 4

Soit la configuration de tâches périodiques indépendantes suivante :

	WCET	D	P
T_1	13	20	20
T_2	8	20	20
T_3	14	20	20
T_4	2	20	20

Cette configuration de tâches est-elle ordonnançable sur deux processeurs ? Justifier.

Systèmes opératoires Temps Réel

Exercice 5

- Dans le système temps réel OSEK, tous les objets sont définis statiquement dans un fichier OIL.
 - L'ordonnanceur permet-il un comportement de type RM ? de type EDF ?
 - L'ordonnanceur permet-il la préemption de tâche ?
- Le système Xtratum est une implantation de la norme ARINC 653 pour le domaine satellite.
 - Qu'est-ce que le TSP et en quoi consiste-t-il ?
 - Pourquoi ce type de solution est-elle adaptée pour les applications critiques telles que celles exécutées dans le domaine avionique ou satellite.
 - Le domaine automobile utilise OSEK. Une solution à base de TSP aurait-elle un intérêt ?
- Le système Linux implante des solutions de ségrégation telles que les CGroups et les Namespaces utilisés notamment par Docker. Son ordonnanceur dispose maintenant de la fonctionnalité SCHED_DEADLINE qui peut être vue comme une implantation de EDF avec réservation de bande-passante. Pensez-vous que Linux peut être utilisé pour exécuter des applications temps réel critiques de la même manière que l'ARINC 653 ?

Exercice 6

Nous souhaitons configurer OSEK pour exécuter les tâches définies dans l'exercice 3. Nous supposons qu'un « timer » génère une interruption de catégorie 2 toutes les millisecondes. La gestion de cette interruption est faite par une routine « hook » qui contrôle l'incrémentation du compteur de « ticks ». Le compteur de « ticks » est configuré de manière à s'incrémenter toutes les millisecondes.

1. Ecrire, en langage OIL, la configuration d'une alarme périodique, A2, associée au compteur *SysTimerCnt* et qui réveille la tâche T2. Les informations concernant la date de démarrage et la période de la tâche T2 sont indiquées à l'exercice 3.
2. Ecrire, en langage OIL, la configuration des tâches T2 en considérant qu'elle possède la priorité la plus faible du système (ce qui n'est peut-être pas le cas du calcul de priorité que vous avez appliqué à l'exercice 3).
3. Quelle(s) fonction(s) doit-on exécuter dans le code de T2 pour accéder à la ressource R1 ?
4. Le problème d'inversion de priorité peut-il survenir en OSEK ?