

Module STR
Examen session 2, 5 juillet 2016
Systèmes temps réel
Durée : 1h20 - 13 points
Tous documents autorisés

Singhoff Frank
singhoff@univ-brest.fr
C-203

NB : ce document contient 2 feuilles recto/verso.

Tâches	Ci	Pi/Di	Priorité
<i>TC1</i>	2	7	0
<i>TC2</i>	5	21	1
<i>TC3</i>	11	42	2

FIGURE 1 – tâches périodiques et critiques

Tâches	Ci	Si	Priorité
<i>TNC1</i>	12	6	3
<i>TNC2</i>	2	7	4

FIGURE 2 – tâches apériodiques et non critiques

Exercice n° 1 : Questions de cours (2 points)

– Question n° 1.1 : 1 point

Expliquer ce qu'on entend par "inversion de priorité". Donner un exemple.

– Question n° 1.2 : 1 point

Décrivez une des solutions généralement utilisées pour éviter ce comportement dans un système temps réel.

Exercice n° 2 : Ordonnancement temps réel (5 points)

– Question n° 2.3 : 1 point

Soit le jeu de tâches périodiques **critiques** de la figure 1. Montrer que ce jeu de tâches est ordonnançable avec RM en mode préemptif **sans calculer le chronogramme d'ordonnement**. On suppose que zéro est le niveau de priorité le plus fort dans le système.

– Question n° 2.4 : 2 points

Calculer l'ordonnement de ces tâches sur la période d'étude.

– Question n° 2.5 : 1 point

On rajoute au jeu de tâches de la figure 1, les tâches apériodiques et **non critiques** de la figure 2. L'ingénieur responsable de la mise en oeuvre de l'application décide que contrairement aux tâches critiques, les tâches apériodiques et non critiques ne seront pas préemptibles. Expliquez pourquoi cette décision est une erreur.

– **Question n° 2.6 : 1 point**

Dessiner le chronogramme d’ordonnancement sur les 30 premières unités de temps en ordonnant **conjointement** les tâches des figures 1 et 2.

Exercice n° 3 : Langage Ada (6 points)

La spécification ci-dessous décrit un type abstrait générique implantant une file d’attente de type FIFO. On rappelle que dans une fifo, les éléments à insérer le sont en queue de la fifo et les éléments à retirer le sont en tête de la fifo.

L’interface ci-dessous est paramétrée par deux informations : la taille maximale de la fifo qui est par défaut de 100 éléments et le type des éléments stockés dans la fifo.

```
generic
  type element is private;
  size : natural := 100;
package fifo is

  procedure inserer_en_queue(e : in element);
  procedure retirer_en_tete(e : out element);

end fifo;
```

– **Question n° 3.1 : 2,5 points**

Dans un premier temps, on vous demande de donner une implémentation du package fifo.

– **Question n° 3.2 : 1 point**

Quels problèmes classiques pose l’accès direct à une instance du package fifo par plusieurs tâches? Vous décrirez un scénario d’ordonnancement qui illustre ces problèmes.

– **Question n° 3.3 : 2,5 points**

- Proposez un programme Ada constitué de deux tâches :
 1. La première insère dans la fifo des éléments de type entiers.
 2. La seconde retire les entiers, puis, les affichent.
- Votre implantation de la fifo devra tenir compte de la question précédente.