

# Simulation d'ordonnancement

## Utilisation de Cheddar

Pour plus de détails, un tutoriel en ligne est disponible ici :

<http://beru.univ-brest.fr/~singhoff/cheddar/contribs/educational/ubo/ETR15/tp.html>

## Exercice

Modéliser les tâches périodiques suivantes définies par les paires (*charge*, *période*) suivantes :

$$\{ (4, 20), (10, 30), (5, 40) \}$$

où une unité de temps vaut 10 millisecondes.

## Question 1

- Lancer `cheddar` sur le serveur
- Décrire un `core` à l'aide du menu `Edit/Hardware/Core`
  - `Name` : le nom unique du composant, par exemple, `core1`.
  - `Scheduler type` : choisir `POSIX_1003_Highest_Priority_First_Protocol`
  - cliquer `close`
- Décrire le CPU à l'aide du menu `Edit/Hardware/Processor`
  - `Name` : le nom unique du CPU, par exemple, `cpu1`.
  - ajouter `core1` à `cpu1` en cliquant `add`
  - cliquer `close`
- Décrire l'espace d'adressage, à l'aide du menu `Edit/Software/Address space`
  - `Name` : le nom unique de cet espace d'adressage, par exemple, `ea`.
  - ajouter `core1` à `cpu1` en cliquant `add`
  - cliquer `add` puis `close`
- Sauvegarder cette configuration sous le nom `mono.xml`
- Décrire les tâches en complétant les champs suivants :
  - `Name` : le nom unique de la tâche : `t1`, `t2`, `t3...`
  - `Task type` : périodique
  - `Capacity` : le WCET de la tâche (temps d'exécution dans le pire des cas).

- **Period** : la période de la tâche.
  - **Deadline** : l'échéance à respecter par la tâche (ici identique à la période)
  - **Priority** : la priorité fixe associée à la tâche.
  - cliquer **add**, saisir les tâches suivantes, puis **close**
- Associer des priorités de type **Rate Monotonic** aux tâches en utilisant le menu **Tools**
  - Tester la simulation en cliquant sur *play*.
  - Sauvegarder la configuration sous le nom **simulation.xml** et l'exporter au format AADL sous le nom **simulation.aadl**

## Question 2

Tester la simulation en écrivant un code Ada correspondant aux 3 tâches ci-dessus. On utilisera l'ordonnancement **FIFO\_Within\_Priorities**. La charge sera simulée en utilisant le package **Ada.Execution\_Time**.

On supposera qu'un double changement de contexte prend au plus 1 milli-seconde, et qu'il est déjà compté dans la charge. Ce qui nous donne les valeurs suivantes en milli-secondes pour les 3 tâches à simuler :

$$\{ (39, 200), (99, 300), (49, 400) \}$$