

# Temps réel mou et Linux

## Déroulement du programme

Le fichier **threads.c** est le fichier contenant le programme principal (avec le fichier de headers correspondant **threads.h**). Les fichiers **tasks.c** et **tasks.h** contiennent les fonctions appelées par les threads et chargées à chaud par les threads.

Lors du parsing du fichier, chaque nouvelle tâche à appeler est enregistrée dans un tableau de *TaskInfo*, structure contenant le nom de la fonction à appeler, le numéro du thread, le numéro du thread suivant, ainsi que la ligne sur la quelle cette tâche se trouve.

Après avoir récupéré toutes les informations relatives aux tâches à exécuter, les threads sont lancés et c'est la fonction *startThread()* qui est donnée en tant que paramètre que *pthread\_create()*.

Le déroulement de la fonction *startThread()* est le suivant :

- Récupération des informations sur la tâche à effectuer (mutex et autres valeurs)
- Boucle principale ( *while(1)* ) :
  - Attente de la libération (puis prise) du mutex avec *sem\_wait*
  - Début du timer si cette tâche est en début de ligne
  - Appel de la fonction à exécuter
  - Fin du timer. Si l'échéance est dépassée, on libère le mutex avec *sem\_post* de la tâche en début de ligne. Sinon, on libère le mutex de la tâche suivante.

## Utilisation du programme

Le fichier **taskList.txt** contient la liste des tâches à effectuer. Il est sous la forme :

```
TASK_NB:<nombre total de tâches>
LINE_NB:<nombre de lignes>
<nb de tâches sur la ligne>:<num. des fcts à appeler>...-END-<échéance de la ligne>
```

Pour lancer le programme, taper :

```
make
./threads.out
```

A noter que le programme est compilé avec les options -Wall et -pedantic.