

## TP 3.

### 1 Objectifs

- Mise en oeuvre de la notion de module en C `.h` et `.c`
- L'exemple choisi est le type file de processus : Interface et réalisation avec une liste chaînée.
- Rappeler le principe de la compilation séparée.
- Introduire l'outil `make` sur cet exemple.

### 2 Travail

1. Compléter l'implémentation du module processus (`processus.c`) en écrivant le code des fonctions<sup>1</sup> présentes dans l'interface (`processus.h`),
2. Compléter le fichier `appli_processus.c` pour tester au mieux vos fonctions (cf. tests unitaires),
3. Rajouter de nouvelles fonctions au module (test de présence d'un processus dans la liste, ajout d'un processus en fin de liste, modification de l'état d'un processus donné de la liste, vider la liste, ...) et les tester.

### Annexe : sous-programmes du module

1. Écrire un sous-programme qui crée une liste vide de processus
2. Écrire un sous-programme qui insère un processus en tête de la liste [on pourra discuter de comment est donné le processus, lié au choix d'implantation du type chaînon]
3. Écrire un sous-programme qui affiche les processus d'une liste
4. Écrire un sous-programme qui enlève un processus de la liste.
5. Écrire un sous-programme permettant d'effectuer une même action sur les états de tous les processus de la liste (par exemple, inverser son état, ou le rendre actif ou le suspendre)

---

1. Ces fonctions ont été vues en partie en CTD.