

# DM – Système de gestion de ressource

Programmation en C

## Introduction

Ce DM est à réaliser en groupe de 3 à 5 personnes. Chaque groupe doit être composé d'obligatoirement une personne ayant déjà développé en langage C.

Le DM est à rendre au plus tard le jour de l'examen du module UTC502.

## Principe

Le travail à réaliser est un ensemble de programmes écrits en C permettant la gestion d'une ressource (par exemple, de la mémoire). La ressource est disponible en quantité limitée, un programme devra donc gérer l'accès (attribution et restitution) à cette ressource lorsque des clients viendront lui en réclamer. Le mécanisme de communication entre le programme principal et les clients est un système client/serveur :

- Le programme principal devra préparer la ressource. La quantité disponible de ressource est transmise en argument à l'exécution. Le programme principal attend ensuite que des clients se connectent. Lorsqu'un client se présentera, le programme devra traiter ses demandes d'accès à la ressource tant qu'il en aura besoin. Le client peut donc réclamer plusieurs fois au programme de lui attribuer une partie des ressources ou de lui en restituer.
- Le programme principal doit être capable de gérer plusieurs clients simultanément.
- Le programme principal affichera à l'écran la quantité de ressource restante, la liste des clients qui sont connectés ainsi que les ressources qu'ils consomment. L'affichage sera rafraîchi à intervalle régulier.
- À l'exécution du programme client, il faudra passer en arguments les éléments lui permettant de joindre le serveur ainsi que les quantités de ressources qu'il devra demander ou relâcher et le délai entre chaque demande.
- Lorsqu'un client demande une attribution de ressources, le programme principal contrôle si la quantité demandée est disponible. Si ce n'est pas le cas, il refuse la demande (pas d'attribution partielle).
- Si un client se déconnecte brutalement, le programme principal libèrera les ressources qu'il lui avait attribuées afin que les autres clients puissent les utiliser.
- Si le programme principal est arrêté, les programmes secondaires s'occupant des clients devront relâcher la connexion avec leur client respectif et libérer les ressources avant de s'arrêter.

Avant de démarrer le sujet, réfléchissez aux différents mécanismes qui permettent aux processus de communiquer entre eux. Il en existe plusieurs qui, combinés ensemble, permettront de répondre à la problématique posée.

## Éléments à rendre

- Un rapport de 15 à 20 pages maximum (hors annexes) dans lequel vous présenterez le fonctionnement des différents programmes réalisés et leur utilité, ainsi que les

mécanismes mis en place pour permettre la communication entre les processus. Le code source sera joint à ce rapport en annexe, ainsi que tout autre élément que vous jugerez utile d'ajouter.

- Les exécutable des programmes C fonctionnels.

## Bonus

Vous avez la possibilité d'ajouter des fonctionnalités bonus à votre programme. Ces fonctionnalités sont à ajouter seulement si l'ensemble des fonctionnalités de base demandées sont présentes. Elles vous permettront d'obtenir des points bonus supplémentaires :

- Possibilité de configurer le serveur et les clients en utilisant un fichier de configuration passé en argument plutôt qu'en donnant les arguments en ligne à l'exécution.
- Possibilité de configurer les clients pour qu'ils fonctionnent de manière entièrement autonome en leur indiquant une plage de quantités de ressource à demander/relâcher ainsi qu'une plage de délai entre chaque demande/relâchement. De cette manière il pourra être possible de créer des profils de clients peu ou très consommateurs de ressource, peu ou très agressifs...
- Possibilité de configurer le serveur en lui donnant des directives supplémentaires pour l'attribution des ressources : un délai minimum entre 2 demandes, une quantité maximale à chaque demande ou au contraire une quantité minimale...