

UNIVERSITÉ DE SHERBROOKE  
DÉPARTEMENT D'INFORMATIQUE

Devoir # 3

**IFT 630**

Processus concurrents et parallélisme

**Devoir à remettre au plus tard le 13 avril 2015**

La correction sera basée sur une démonstration du bon fonctionnement de vos programmes. Vous devez me soumettre votre programme à l'aide de turnin avant la date de remise. Une démonstration devra ensuite être prévue.

Planter un serveur en utilisant le concept de multiples fils d'exécution (multithread) de Solaris ou de Posix. Pour utiliser le concept de fils d'exécution, vous devez utiliser les langages C ou C++. Vous devez utiliser la communication par messages de Unix (ou les socket) pour faire communiquer les clients avec le serveur. Vous trouverez des fonctions permettant d'utiliser la communication par messages sur le site WEB du cours. Vous trouverez des informations sur l'utilisation des bibliothèques de fils d'exécution de la STL, de Posix et de Solaris sur le WEB.

Le serveur que vous devez planter devra être un serveur de transfert de fichiers ou un autre type de serveur permettant de faire ressortir les avantages des multiples fils d'exécution (multi-threads). Pour montrer son bon fonctionnement et le parallélisme présent dans le serveur, vous devez créer plusieurs clients (processus et non des fils d'exécution) qui feront des demandes en parallèle. Votre serveur doit être fiable, i.e. il devra toujours terminer proprement même en présence d'erreurs irrécupérables. Pour y parvenir, vous devez utiliser les « signaux » de Unix.

Pour le démarrage, vous devez utiliser un processus principal qui utilisera les commandes « *fork* » et « *exec* » pour démarrer tous les clients et le serveur (vous trouverez une bibliothèque C++ pour manipuler les processus sur la page WEB du cours). Ce même serveur doit se charger de bien synchroniser la fin du système.

Il devra y avoir un client démarré interactivement afin de vérifier la bonne capture des signaux. Vous devez prévoir une fenêtre d'exécution assez longue pour tester la capture des signaux sur le serveur.