

Programmation système avancée

TP n° 10 : Encore des threads

Exercice 1 :

Dans le fichier `mandelbrot.tar.gz`, vous trouverez un programme qui permet d'explorer une approximation de l'ensemble de Mandelbrot.

1. Compilez le programme fourni et jouez avec. Les boutons gauche et droit permettent de changer l'échelle. Le temps de calcul s'affiche dans le terminal. Constatez que les parties qui se trouvent dans l'ensemble demandent un temps de calcul beaucoup plus élevé que les parties qui en sont éloignées.
2. Modifiez le programme pour que le calcul s'effectue sur tous les cœurs de votre machine. Vous pouvez en déterminer le nombre à l'aide de `sysconf(_SC_NPROCESSORS_ONLN)`. Attention, il n'est pas légal de faire des appels aux bibliothèques *Gtk* et *Cairo* depuis des *threads* autres que celui qui exécute la fonction `draw_cb`, il faudra donc faire le *rendering* après un appel à `pthread_join`.
3. Le partage de charge se fait-il bien lorsque l'ensemble couvre une moitié de la fenêtre ? Ce problème est-il facile à résoudre ? Quel rapport avec le projet ?

Exercice 2 :

Dans le TP 7, vous avez implémenté les *mutex* et les variables de condition à l'aide de sémaphores.

1. Quels champs de la structure `motd` implémentent un *mutex* ? Une variable de condition ?
2. Quelles opérations implémente chacune des fonctions de l'exercice 2 du TP 7 ?
3. Modifiez le code du TP 7 pour qu'il utilise un `pthread_mutex_t` et une `pthread_cond_t`. Comme ces structures seront partagées entre processus, il faudra spécifier l'attribut *pshared* (voyez `pthread_mutexattr_setpshared` et `pthread_condattr_setpshared`).