Dias Alain M2

Rapport TP1

Pour la réalisation de ce TP, j'avais estimé le temps de travail nécessaire à 5 heures alors qu'il m'a fallu réellement 8 heures.

Je n'ai pas réussi à atteindre cet objectif suite à divers problèmes durant l'implémentation de mon algorithme de remplissage du tableau en colimaçon.

Initialement j'ai essayé de réaliser cet exercice à l'aide de tableau à deux dimensions ainsi qu'en calculant la direction de remplissage tout au long de mon parcours de tableau. Ces choix n'ont pas été gardés. En effet, cet algorithme utilise finalement un tableau à une dimension.
En ce qui concerne mon choix de calcul de direction durant le remplissage du tableau, j'ai préféré opter pour une solution plus simple consistant a remplir les quatre bords directement. En effet, ma première solution étant difficile à déboguer, j'ai du revoir ma conception pour une solution plus simple.

Mon programme ne pourrait pas directement être parallélisé et fonctionner dans un programme multi-threads puisqu'il ne s'arrête que lorsque le tableau est complètement remplis. Il serait cependant facile de le rendre parallélisable en concevant un algorithme fonctionnant sur le même principe que le mien. En effet, en s'appuyant sur un algorithme remplissant le périmètre d'un niveau du tableau, on pourrait confier la tache de remplir plusieurs niveaux à plusieurs threads.

Pour evaluer le temps d'execution de mon programme j'utilise « perf stat » :

Tableau 10x10	: 0,000908960 seconds
Tableau 100x100	: 0,001671653 seconds
Tableau 1000x1000	: 0,081716581 seconds
Tableau 10000x10000	: 8,004847092 seconds

Note : Mon programme utilise une librairie .so pour l'execution des programmes de tests ainsi que du main. Il faut exécuter dans le terminal « export LD_LIBRARY_PATH=./lib: $LD_LIBRARY_PATH » pour que les programmes fonctionnent.