ENSG Géomatique

ÉCOLE NATIONALE
DES SCIENCES
GÉOGRAPHIQUES

Rapport du TP 3 Programmation Parallèle

Ce TP reprend ce qui a été fait dans les deux premiers TP mais en utilisant la bibliothèque OpenMP en langage C.

Réalisé par : Amina Aiache
Elève ingénieur en 3ème année

Encadré par : M. Ahmad AUDI

Table des matières

Introduction

Ce TP reprend ce qui a été fait dans les deux premiers TP mais en utilisant la bibliothèque OpenMP en langage C.

Nous a avons utilisé le compilateur GCC qui permet de paralléliser des codes contenant des directives OpenMP.

OpenMP

Question 1

Les ordinateur fixes que nous avons utilisés, possèdent un Processeur :

Intel (R) Core (TM) i7-3770 CPU @ 3.4 GHz

Et donc ils ont 8 processeurs.

Question 2

Pour tester l'environnement, j'ai fais un test simple qui affiche un « Hello World » depuis chaque processus (code source TestOpenMP/test_omp.c). Le résultat de l'exécution est :

Question 3

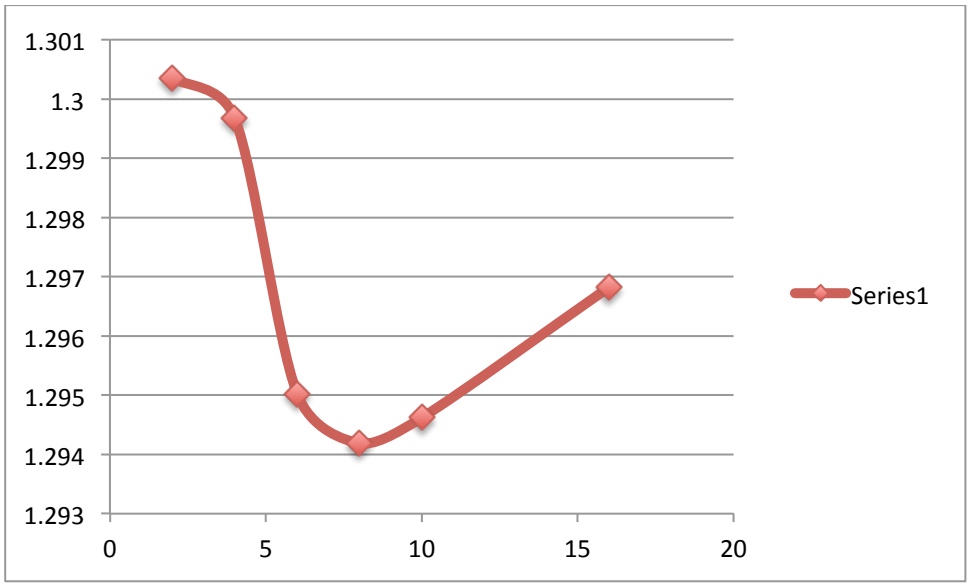
Pour analyser la performance du code de calcul de l'ensemble de Mandelbrot, j'ai comparé le temps d'exécution du code en augmentant le nombre de processus avec deux exemples :

- Les paramètres par défaut :

Le tableau suivant montre les résultats :

nombre de processus	temps d'exécution
2	0,644157
4	0,642903
6	0,642226
8	0,642161
10	0,642454
16	0,642636

Voici le graphe qui montre l'évolution :

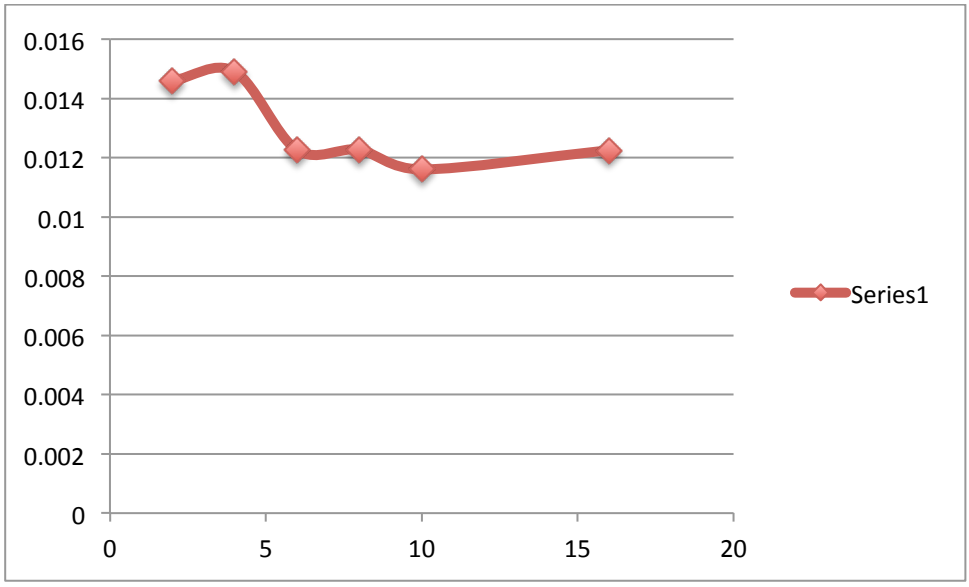


Ce qui confirme la question 1, l'ordinateur n'a que 8 cœurs logique, et donc au-delà de 8 processus le temps d'exécution de diminue pas.

- mandel_omp 800 800 -1.5 -0.1 -1.3 0.1 10000 :

nombre de processus	temps d'exécution
2	1,30035
4	1,29968
6	1,29503
8	1,29419
10	1,29462
16	1,29682

Le graphe :

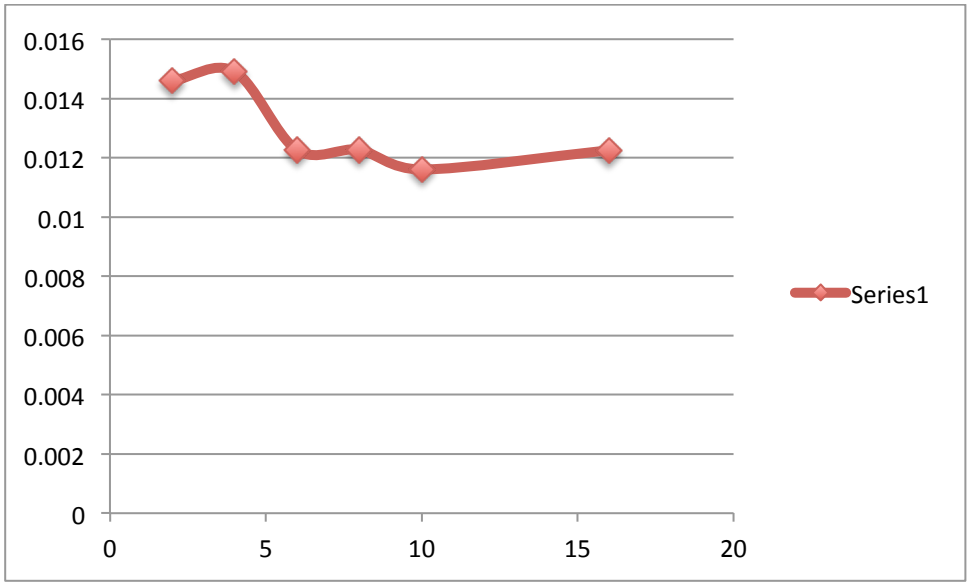


Question 4

Pour analyser la performance du code de l'application du filtre de convolution à une image, j'ai comparé le temps d'exécution du code parallèle en augmentant le nombre de processus, le tableau suivant montre les résultats obtenus :

nombre de processus	temps d'exécution
2	0,0145988
4	0,0149109
6	0,01226
8	0,0122712
10	0,011606
16	0,012253

Le graphe ci-dessous montre l'évolution du temps d'exécution en fonction du nombre de processus :



Pareil que pour la question suivante, on constate que le temps d'exécution diminue, mais dès qu'on atteint 8 processus, il reste constant ou au contraire commence à croitre.