TP4

Objectif

Initiation à la programmation GPU avec cuda

Travail à réaliser

1- Copier depuis ~hamrouni le dossier cudasamples (cp -r ~hamrouni/cudasamples .) dans votre dossier de travail.

A- Compiler le programme deviceQuery (nvcc -I. deviceQuery.cpp [-o …] :
Exécuter et noter les éléments évoqués en cours et en TD.

B- Faire de même avec le programme bandwidthTest. Noter les résultats.

3- Prendre une copie du squelette du programme mul_matG1B1-squel.cu depuis moodle, et le compléter pour réaliser la multiplication de 2 matrices sur GPU.

A- Tester le et vérifier les résultats et le temps d'exécution pour différentes valeurs de BLOCK_SIZE.
Que se passe-t-il avec TM > 1024 ? Pourquoi ?
B- Inverser les formules de i et j. Que constate-t-on ? Pourquoi ?
C- Ajouter l'affichage du temps de transfert de A et B vers le GPU et celui de C vers le CPU. Ces temps sont-ils cohérents avec les résultats du 1-B ? Quel est le rapport du coût de transfert / temps de multiplication ?

4- Transformer ce programme de manière à pouvoir augmenter TM tout en gardant des blocs à une dimension
- Tester le et vérifier les résultats pour TM=2048
- Faire varier BLOCK_SIZE_X pour TM=2048 et vérifier les résultats
- Faire varier TM pour BLOCK_SIZE_X = 512 et vérifier les résultats

5- Transformer ce programme en utilisant une grille à deux dimensions :
(dim3 grid(GRID_SIZE_X, GRID_SIZE_Y) et des blocs à deux dimensions (dim3 block(BLOCK_SIZE_X, BLOCK_SIZE_Y))
- Tester et vérifier les résultats pour TM=2048
- Faire varier BLOCK_SIZE_X et BLOCK_SIZE_Y : 32-32, 16-16, 32-16, 16-32, 64-8, 8-64, 128-4, 4-128, 128-2, 2-128. Que constate-t-on ?
- inverser les formules de calcul de i et j, et faire les mêmes tests. Que constate-t-on ?

Au final, quelle configuration donne les meilleures performances?