

TD N°5 : Programmation des GPUs NVIDIA avec CUDA

Exercice 1

Ecrire un programme **CUDA C** qui permet à un **GPU NVIDIA** d'additionner deux vecteurs de taille $N = 1000$ en mobilisant plusieurs threads appartenant à différents blocs.

Exercice 2

Ecrire un programme **CUDA C** qui permet à un **GPU NVIDIA** de réaliser le produit scalaire de deux vecteurs, tel que :

1. Les vecteurs sont de taille $N = 512$ et leur produit scalaire doit être réalisé par des threads appartenant au même bloc.
2. Les vecteurs sont de taille $N = 2048$ et leur produit scalaire doit être réalisé par des threads appartenant à différents blocs.

Exercice 3

Ecrire un programme **CUDA C** qui permet à un **GPU NVIDIA** d'inverser un vecteur de taille $N = 2048$. Autrement dit, étant donné un vecteur $\{a_0, a_1, \dots, a_{N-1}\}$, le **GPU** donne en sortie le vecteur $\{a_{N-1}, a_{N-2}, \dots, a_0\}$. Ceci doit être réalisé en mobilisant plusieurs threads appartenant à différents blocs.

N.B. Pour les exercices 1, 2, et 3, le bloc peut contenir jusqu'à 512 threads (16 *warps* où le *warp* représente un groupe de 32 threads pouvant s'exécuter simultanément sur un multiprocesseur du **GPU**). Aussi, le bloc est organisé en une dimension.

Exercice 4

Ecrire un programme **CUDA C** qui permet à un **GPU NVIDIA** de calculer le produit scalaire de deux vecteurs de taille 128 en mobilisant plusieurs threads appartenant à un seul bloc. Il est à noter que dans ce cas, la taille du bloc 2D est égale à 16×16 .