

Exercice 1. Calcul de Π

Pour calculer Π on se propose d'utiliser la méthode suivante. On considère un disque de rayon 1 centré sur (0,0) et on parcourt la surface du carré de côté 2 dans lequel est inscrit le cercle. Pour chaque point on teste si ce point est dans le disque ou non ($x^2 + y^2 \leq 1$). L'aire du carré est 4 et celle du disque est Π . Par conséquent si on a n_1 points à l'intérieur du disque sur n_2 points testés, $\Pi \approx \frac{4 \times n_1}{n_2}$.

1. Écrire un programme OpenCL qui effectue l'approximation de Π suivant cette méthode et en utilisant les working groupes pour éviter trop d'accès à la mémoire globale.
2. Comparer les performances avec un programme utilisant des directives OpenMP pour partager le calcul.

Exercice 2. Diagramme de fréquences

On considère un vecteur de float *vec* ne contenant que des nombres compris dans l'intervalle $[0, N[$. On veut obtenir un tableau *freq* de taille N tel que $\forall i \in [0, N[$ *freq*[*i*] est le nombre de valeurs de *vec* comprises dans l'intervalle $[i, i + 1[$.

1. Écrire un programme OpenCL permettant d'effectuer ce calcul, en considérant $N = 100$ par exemple.
2. Optimiser ce programme afin d'éviter les opérations atomiques sur la mémoire globale.
3. Comparer les performances y compris avec une version CPU avec des directives OpenMP.

Exemple pour un tableau de fréquences avec $N = 3$

vec = [1.0, 2.3, 0.3, 2.3, 0.8, 2.5, 0.4, 0.3, 0.1], *freq* = [5, 1, 3]

Exercice 3. Flots d'accumulation

Le point de départ de ce calcul est un modèle numérique de terrain (MNT), c'est-à-dire une matrice 2D représentant un terrain pour laquelle chaque valeur indique la hauteur du terrain en ce point.

A partir du MNT nous pouvons calculer la matrice des directions. Cette matrice indique en chaque point, la direction vers laquelle une goutte de pluie se dirigerait si elle tombait en ce point. Cette direction est celle de la plus forte pente entre le point considéré et ses voisins. Pour coder les directions on peut prendre la convention ci-dessous (*aucune* indique que le point considéré n'a pas de voisin plus bas que lui):

Direction	NO	N	NE	E	SE	S	SO	O	aucune
Code	1	2	3	4	5	6	7	8	0

Une fois la matrice des directions calculée, on peut calculer une nouvelle matrice permettant de connaître les flots d'accumulation. Chaque valeur de cette matrice représente pour le point du MNT correspondant, le nombre de points du MNT qui s'écoulent vers lui. Ce calcul peut se faire de manière itérative, de la manière suivante:

- Initialiser la matrice à 0
- Tant qu'on n'a pas la valeur d'accumulation de tous les points :
 - > si pour un point donné on connaît la valeur d'accumulation de tous ses voisins qui s'écoulent vers lui alors mettre la somme des valeurs d'accumulation des voisins +1 dans le point considéré.

Finalement, à partir des flots d'accumulation on peut trouver le lit des cours d'eau en appliquant un algorithme de seuillage. Si une cellule a une valeur d'accumulation supérieure à un seuil, c'est un cours d'eau, sinon c'est un terrain sec.

78	72	69	71	58	49
74	67	56	49	46	50
69	53	44	37	38	38
64	58	55	22	31	24
68	51	47	21	16	19
74	53	54	12	11	12

Modèle Numérique de Terrain (MNT)

↘	↘	↘	↘	↓	↙
↘	↘	↘	↓	↙	↓
→	→	↘	↓	↙	↓
↘	↗	↘	↘	↓	↙
→	→	↘	↘	↓	↙
↗	↗	→	→		←

Directions

1	1	1	1	1	1
1	2	2	2	4	1
1	3	7	9	1	2
1	1	1	18	1	3
1	4	6	2	23	1
1	1	1	8	36	1

Flots d'accumulation

Figure 1: Les différentes matrices

1. Ecrire un programme permettant d'effectuer ces différentes étapes sur carte graphique et optimisant au maximum les algorithmes utilisés
2. Montrer les performances de votre implémentation en la comparant avec une version CPU (utilisant éventuellement openMP).