Questions à Choix Multiples / Multiple Choice Questions

Question 1:

Quelle sera la valeur du tableau a[] après l'exécution du noyau suivant avec configuration <<<3, 4>>> ? / What will be the value of array a[] after executing the following kernel with configuration <<<3, 4>>>?

```
__global__ void kernel(int *a) {
   int i = threadIdx.x + blockIdx.x * blockDim.x;
   a[i] = blockDim.x;
}
```

Options / Choix:

a) [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

b) [0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]

c) [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2]

d) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Réponse / Answer: a) [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

Question 2:

Quelle sera la valeur du tableau a[] après l'exécution du noyau suivant avec configuration <<<3, 4>>> ? / What will be the value of array a[] after executing the following kernel with configuration <<<3, 4>>>?

```
__global__ void kernel(int *a) {
   int i = threadIdx.x + blockIdx.x * blockDim.x;
   a[i] = threadIdx.x;
}
```

Options / Choix:

a) [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

b) [0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]

c) [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2]

d) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Réponse / Answer: b) [0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]

Question 3:

Quelle sera la valeur du tableau a[] après l'exécution du noyau suivant avec configuration <<<3, 4>>> ? / What will be the value of array a[] after executing the following kernel with configuration <<<3, 4>>>?

```
__global__ void kernel(int *a) {
   int i = threadIdx.x + blockIdx.x * blockDim.x;
   a[i] = blockIdx.x;
}
```

Options / Choix:

a) [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

b) [0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]

c) [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2]

d) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Réponse / Answer: c) [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2]

Question 4:

Quelle sera la valeur du tableau a[] après l'exécution du noyau suivant avec configuration <<<3, 4>>> ? / What will be the value of array a[] after executing the following kernel with configuration <<<3, 4>>>?

```
__global__ void kernel(int *a) {
   int i = threadIdx.x + blockIdx.x * blockDim.x;
   a[i] = i;
}
```

Options / Choix:

a) [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

b) [0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3]

c) [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2]

d) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Réponse / Answer: d) [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]