NPTEL Online Certification Courses Indian Institute of Technology Kharagpur

GPU Architectures and Programming
Assignment- Week 4

TYPE OF QUESTION: MCQ/MSQ

Number of questions: 10 **Total mark: 10 X 1 = 10**

MCQ / MSQ Question

Question 1:

Consider a 1D CUDA kernel that computes on a 1D array A of size 2048 with launch parameters <<<(32),(64)>>> . Each CUDA thread operates only on one data point. On what data point will the thread with threadIdx.x=20, blockIdx.x=20 operate on. Assume that thread ids along the x-dimension represent rows and thread ids along the y-dimension represent columns. Choose the correct option.

A. A[200]
B. A[40]
C. A[1300]
D. None of the above

Ans: C
Solution:
Grid configuration:<<<32,1,1>>>
This means there are 32 blocks along x direction
Block configuration: <<<64,1,1>>>
This means there are 64 threads along the x-dimension(1D)
Total threads = 64X32=2048, this matches the size of A

Global thread Id = threadIdx.x + blockIdx.x * blockDim.x
Global thread Id= 20 + 64*20 = 20 + 1280 = 1300
So it will operate on data point = A[1300]

Question 2:

Consider a kernel launched with the following launch configuration parameters.

<<<(2,2,1),(32,32,4)>>>

What is the total number of CUDA threads that are spawned for the given launch configuration?

A. 16384
B. 2048
C. 8192
D. 4096

Ans: A
Solution:

Total number of threads: 2*2*1*32*32*4 = 16384

Question 3:

For the launch parameter configuration provided in Question 2, what are the minimum and maximum values of threadIdx.x?

A. min=0, max=1023
B. min=1,max=32
C. min=0,max=31
D. min=1,max=1024

Answer: C

Solution:
We can see that blockDim.x = 32 i.e the number of threads in in x dimension of block is 32. So id of the threads in the block will range from 0 to 31.

Question 4:

For the launch parameter configuration provided in Question 2, what are the minimum and maximum values of blockIdx.y?

A. min=0,max=1
B. min=1,max=2
C. min=0,max=31
D. min=1,max=32

Answer: A

Solution:

We can see that blockDim.y = 32 i.e the number of threads in in y dimension of block is 32. So id of the threads in the block will range from 0 to 31.

Question 5:

A CUDA kernel is launched with a following grid and block configuration. Given:

Grid configuration: <<<2, ?, 1>>>

Block configuration:<<<4, 4, ?>>>

With what values should ? be replaced if the total number of threads are 128:

A. Grid: <<<2, 4, 1>>>, Block: <4, 4, 2>
B. Grid: <<<2, 4, 1>>>, Block: <4, 4, 1>
C. Grid: <<<2, 2, 1>>>, Block: <4, 4, 2>
D. Grid: <<<2, 8, 1>>>, Block: <4, 4, 2>

Ans: B

Solution:

Total number of threads 2*4*4*4 = 128

Question 6:

What will happen if `__syncthreads()` is called conditionally, as shown in the code snippet :

```
__global__ void conditionalSync() {
    if (threadIdx.x % 2 == 0) {
        __syncthreads();
    }
}
```

A. All threads synchronize correctly.
B. Only even threads synchronize correctly.
C. Kernel execution may lead to undefined behavior.
D. No synchronization takes place.

Ans: C, D

Solution:

__syncthreadds() function must be called by all the threads in the block. But here threads with only even id calls the synchronization function. This leads to undefined behavior.

Question 7:

Consider the given kernel code:

```
__global__ void syncExample(int *array) {
    int idx = threadIdx.x + blockIdx.x * blockDim.x;
    if (threadIdx.x == 0) {
        array[idx] = 1;
    }
    __syncthreads();
    array[idx] += threadIdx.x;
}
```

If the kernel is launched with<<<1, 4>>> and array is initially {0, 0, 0, 0}, what is the value of array after execution?

A. {1, 2, 3, 4}
B. {1, 1, 1, 1}
C. {1, 2, 3, 4}
D. {1, 1, 2, 3}

Ans: D

Solution:

if(threadIdx.x == 0) condition is true only for thread 0. Thus array[0] is set to 1.
__syncthreads() function ensures that all threads wait until array[0] is updated and then all thread execute array[idx] += threadIdx.x.

threadIdx.x = 0: array[0] = 1+0 =1
threadIdx.x = 1: array[1] = 0+1 =1
threadIdx.x = 2: array[2] = 0+2 =2
threadIdx.x = 3: array[3] = 0+3 =3

Question 8:

A CUDA kernel with a 3D grid and block configuration is launched. Let total number of blocks be 72 and total number of threads be 576. Then what can be the difrent ways to represent a block structure.

A. (2, 2, 2)
B. (3, 2, 3)
C. (2, 4, 1)
D. (1, 8, 0)

Ans: A, C

Solution:
Threads per block = 576 / 72 = 8
so, we can see that 2*2*2=8 and 1*8 =8

Question 9:

Consider a 2D CUDA kernel that computes on a 2D matrix M of size 2048x2048 with launch parameters <<<(32,32),(64,64)>>> . Each CUDA thread operates only on one data point. On what data point will the thread with the threadIdx.x=0, threadIdx.y=0, blockIdx.x=10 and =blockIdx.y=10 operate on. Assume that thread ids along the x-dimension represent rows and thread ids along the y-dimension represent columns. Choose the correct option.

A. M[10][10]
B. M[32][32]
C. M[640][640]
D. None of the above

Ans: C

Solution:

Total number of thredas in the grid is:
Thread per block X Blocks in grid = (64 X 64) x (32 X 32) = 2048 X 2048, This is equal to the size of the matrix M.
Global Row(x) = blockIdx.x X blockDim.x + threadIdx.x
Global Row(x) = 10 X 64 + 0 = 640

Global Column(y) = blockIdx.y X blockDim.y + threadIdx.y = 10 X 64 +0 = 640
So thread operates on data point M[640][640]

Question 10:

Which among the following statement is FALSE:

A. gridDim.x,y,z gives the number of blocks in a grid, in the x,y,z direction
respectively.
B. blockDim.x,y,z gives the number of threads in a block, in the x,y,z direction respectively
C. blockDim.x * gridDim.x gives the number of threads in a grid in the x direction
D. Each thread in a block has unique id given by system variable gridIdx.x

Ans: D

***********END*************