Assignment 5

The due date for submitting this assignment has passed. **Due on 2020-03-04, 23:59 IST.**
As per our records you have not submitted this assignment.

1) Consider a GPU architecture with the following constraints.

The total number of threads in a thread block is 512. The maximum number of threads in the x-dimension is 256, in the y-dimension is 256 and in the z dimension is 8. Considering a three dimensional CUDA kernel, which of the following thread block configurations is feasible?

A. `blockDim.x = 256, blockDim.y = 256, blockDim.z = 8`
B. `blockDim.x = 32, blockDim.y = 32, blockDim.z = 8`
C. `blockDim.x = 32, blockDim.y = 16, blockDim.z = 8`
D. `blockDim.x = 32, blockDim.y = 16, blockDim.z = 1`

No, the answer is incorrect.
Score: 0
Accepted Answers:
d.

2)
Which of the following statements is false?

A. Multiple CUDA thread blocks can execute in a single SM of an NVIDIA GPU.
B. The computation of one CUDA thread block can be distributed across multiple SMs.
C. Threads across multiple CUDA thread blocks cannot be synchronized using `syncthreads()`.
D. Threads in a single CUDA thread block mapped to a single SM can communicate using shared memory.

- a. 
- b. 
- c. 
- d. 

No, the answer is incorrect.
Score: 0
Accepted Answers: 
- b.

3)

Consider a GPU architecture where the warp size is 16 and a kernel program which is launched with a configuration where the total number of threads in a thread block is 32. Consider the following conditional statements in the kernel.

- i. if(tidx.x < 16)
- ii. if(tidx.x % 16)
- iii. if(tidx.x % 32)
- iv. if(tidx.x / 2 > 0)

Which of the following operations is correct?

A. All conditional branches i-iv are divergent
B. Conditional branches i, ii and iii are NOT divergent
C. Conditional branches ii and iv are divergent
D. Conditional branch i is divergent

- a. 
- b. 
- c. 
- d. 

No, the answer is incorrect.
Score: 0
Accepted Answers: 
- b.
Consider a kernel processing a 2D matrix of dimensions 2048 x 2048 where each thread is assigned to perform an operation on a single element of the matrix. The kernel is launched with the following grid and block configurations: \( <a,32,2> \) blocks of \( <64,b,2> \). What are the values of \( a \) and \( b \)? Select the correct operation from below.

A. \( a=32 \), \( b=16 \)
B. \( a=64 \), \( b=6 \)
C. \( a=16 \), \( b=64 \)
D. None of the above

No, the answer is incorrect. Score: 0
Accepted Answers: d.

5) Consider a kernel processing a 1D array of 8192 elements where each thread is assigned to perform an operation on a single element of the array. The kernel is launched with the following grid and block configurations: \( <16,a,2> \) blocks of \( <b,2,4> \)

A. \( a=16 \), \( b=2 \)
B. \( a=2 \), \( b=16 \)
C. \( a=8 \), \( b=4 \)
D. All of the above

No, the answer is incorrect. Score: 0
Accepted Answers: d.

6)
Consider the following kernel processing A which is a 1D array of 8192 elements on a GPU architecture where the warp size is 8. The kernel is launched with a configuration of \(<128,1,1>\) blocks of \(<32,1,1>\) threads. Assume each element of A is 0 initially.

```c
#define WARP_SIZE 8
__global__ void kernel1(float *A)
{
    int tid = threadIdx.x;
    int gid = blockIdx.x * blockDim.x + threadIdx.x;
    int loop_bound = threadIdx.x / 8;
    if(tid/8)
    {
        for(i=0;i<loopbound;i++)
        {
            A[gid]++;
        }
    }
}
```

After the execution of the above program, what would be the value of A[8191]?

A. 16
B. 8
C. 9
D. 3

No, the answer is incorrect.
Score: 0
Accepted Answers: d.
Consider the following kernel snippet.

```c
__global__ void kernel1(float *A)
{
    int tid = threadIdx.x;
    int gid = blockIdx.x * blockDim.x + threadIdx.x;
    int loop_bound = threadIdx.x / WARP_SIZE;
    if(tid/WARP_SIZE)
    {
        for(i=0;i<loopbound;i++)
        {
            A[gid]++;
        }
    }
}
```

Consider three different GPU architectures - A1 where WARP_SIZE is 8; A2 where WARP_SIZE is 16 and A3 where WARP_SIZE is 32. Match and pair the following columns of statements.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>i. A1</td>
<td>a. kernel exhibits divergent behaviour</td>
</tr>
<tr>
<td>ii A2</td>
<td>b. kernel does not exhibit divergent behaviour</td>
</tr>
<tr>
<td>iii. A3</td>
<td></td>
</tr>
</tbody>
</table>

Select the correct option from the following.

A. i -> a, ii -> b, iii -> b 
B. i -> b, ii -> a, iii -> b 
C. i -> b, ii -> b, iii -> b 
D. i -> a, ii -> b, iii -> a 

No, the answer is incorrect.
Score: 0
Accepted Answers:
c.

8) 12 points
Consider the following kernel snippet executing on a GPU architecture where the warp size is 8. Assume that the number of threads in a thread block at the time of launching the kernel is 32.

```c
__global__ void kernel1(float *A)
{
    int tid = threadIdx.x;
    if(tid%2 != 0)
    {
        if(tid%3 == 0)
        {
            A[tid]++;
        }
    }
    else
    {
        A[tid]--;
    }
}
```

Match and pair the following.

i. thread ids 3, 9, 15, 21  
a. performs increment operation
ii. thread ids 2, 4, 6, 8 
b. performs decrement operation
iii. thread ids 1, 5, 7, 11  
c. does no work

Select the correct option from below

A. i->b, ii->a, iii->c
B. i->a, ii->b, iii->c
C. i->c, ii->a, iii->b
D. i->a, ii->c, iii->b

No, the answer is incorrect.
Score: 0
Accepted Answers: 
b.
Consider again the kernel snippet in the above question. Let us define warp utilization as
the percentage of threads active in a given warp. The warp utilization differs for
each warp instance in a block of threads in the above program. What is the
minimum and maximum warp utilization factors in the above program, considering
an architecture where the warp size is 8 and the thread block size at the time of
launching the kernel is 32?

Select the correct option from below
   A. min = 12.5%, max = 50%
   B. min = 12.5%, max = 25%
   C. min = 25%, max = 25%
   D. min = 25%, max = 75%

   a.
   b.
   c.
   d.

No, the answer is incorrect.
Score: 0
Accepted Answers: b.

10) 12 points
Consider the following kernel snippet in the above question.

```c
__global__ void kernel1(float *A)
{
    int tid = threadIdx.x;
    if(tid%2 != 0)
    {
        if(tid%3 == 0)
        {
            A[tid]++;
        }
        else
        {
            A[tid]--;
        }
    }
    else
    {
        A[tid]--;
    }
}
```

The increment and decrement operations take 2 cycles each to execute. Assume all other operations (assignment, conditional statement evaluation, memory accesses) take no time to execute. Given this, what is the total execution time for a thread block (in cycles) for an architecture where warp size is 16.

Select the correct option from below
A. 2  
B. 6  
C. 9  
D. 4  

No, the answer is incorrect.
Score: 0
Accepted Answers: 
b.