Concurrent Programming

Lab 1 Answers Q1 to Q5

1)

a) Quad-core processor

$$Latency = 50\ ns$$

$$Throughput = \frac{4}{50} = 0.08\ elements/ns$$

b) GPU with 480 cores

$$Latency = 1.6 * 50 = 80\ ns$$

$$Throughput = \frac{480}{80} = 6\ elements/ns$$

2)

a) __global__ - “Declaration specifier”, this is the way that cuda knows this code is kernel as opposed to CPU code.
b) cudaMemcpy( ) – CPU copies input data from CPU to GPU or CPU copies results from GPU to CPU
c) cudaMalloc( ) – CPU allocates the memory on GPU
d) blockDim( ) – Gives the size of a block

3) kernel<<< dim3(8, 4), dim3(8, 16)>>>(...);

The first parameter dim3(8, 4) is the dimensionality of the grids of the block. In here total 32 blocks.

The second parameter dim3(8, 16) specify the each one of those blocks. That’s the number of threads in each block. 128 threads per block

Totally 32*128 = 4096 threads will be created

4)

Map - Map is one to one pattern. In map pattern we can do same function or computation on each piece of data. Map is very efficient on GPUs.

Gather – Gather is many to one pattern. Gather is different from map in a way that reads/ computes average value from several locations and write them into single piece of output. Image blur operation is good example for gather pattern.

Scatter – Scatter is one to many pattern. In scatter rather than calculate the average of the input, each thread takes an input value from location and writes fractions of the value to several output locations. It is opposite of the gather pattern.

Reduce – Reduce is all to one pattern. Reduce is do computation using all input location and writes in one location.

5) Advantages of the automated mappings

The first advantage is that hardware can runs things really efficiently, because it has so much flexibility. There is no need to waiting for any others to complete because if one thread blocks complete quickly then SM can immediately schedule another thread block.

The biggest advantage is scalability. That means we can scale all the way down to a GPU that would be running with single SM.

Disadvantages

We can make no assumptions about what blocks will runs on what SM and also we can't have any explicit communications between blocks. This make to happen something called deadlock in parallel computing.