GPU examples

1. (a) GPUs have very high internal memory bandwidths, considerably higher than CPUs. Why is this bandwidth still a bottleneck on GPUs? How can applications work around this bottleneck?

(b) What is memory coalescing? What is the difference between coalesced memory access on a GPU, and a traditional multi-core CPU access pattern? (Please draw a picture). Why are coalesced memory accesses faster than non-coalesced accesses on GPUs?

2. Let APU (accelerator processing unit) be an accelerator architecture with 32 dual-threaded cores, running at 1 GHz. The local memory of APU has a bandwidth of 32 GB/s. The bandwidth of the connection between the APU local memory and the system memory (e.g., PCI-express) is 8 GB/s.

Also consider a collection of $M$ images, each having $N \times N$ pixels, encoded using 1 float (4 bytes) per pixel, and the following generic filtering application:

1. forall(currentImage=1..M)
2. forall(i=1..N)
3. forall(j=1..N)
4. OutputImage[i,j] = Process(InputImage[i-1,j], InputImage[i,j], InputImage[i+1,j]);

Note that the operation $Process$ executes 4 floating point operations, and assume the images are correctly bordered such that no out-of-bounds errors appear. The CPU runs the application with an average performance of 4 GFLOPs. In an attempt to improve performance, we consider offloading lines 2-4 as an APU kernel. The kernel is able to use half of the peak bandwidth of the local APU memory.

Answer the following questions:

(a) What is the peak performance of APU?

(b) What is the AI of the kernel? Is the kernel memory-bound or compute-bound? Justify your answer.
(c) What is the achieved performance (in GFLOPs) of the kernel? What is the expected execution time (per image)? What is the speed-up versus the CPU (per image)?

(d) Consider the complete accelerated application (CPU and APU, including the back-and-forth transfer of the images). For what values of N does offloading increase the overall application performance? Justify your answer.

(e) Is there any way double-buffering can be used to increase application performance? How?