GPU

- GPU are accelerators that supplement a CPU, so they do not need be able to perform all the tasks of a CPU

- GPU problems sizes : hundreds of mega bytes to giga bytes.

- GPUs rely on hardware multi-threading to hide the latency to memory
  CPUs " multilevel caches to overcome the long latency to memory.
  메모리에서 데이터를 가져올 때 그것과 관련없는 수백개의 threads를 돌림.

- GPU memory bandwidth > latency.

- GPU는 각 프레임마다 수많은 thread를 동적으로 생성하고 관리 및 스케줄링 하기에 많다.

GPU hardware has two levels of hardware schedulers
1. The thread Block Scheduler that assigns blocks of threads to multi-threaded SIMD processors, and
2. the SIMD Thread Scheduler within a SIMD processor, which schedules when SIMD threads should run.

Local memory! Shared by SIMD Lanes within multithreaded SIMD processor. But this memory is not shared between multithreaded SIMD processors.

GPU memory! off chip DRAM shared by the whole GPU and all thread blocks

- 전체 application을 위한 많은양 (large ~~cha~~ caches ✕
  smaller streaming caches and rely on extensive
  multi-threading of threads of SIMD instructions. to hide
  the long latency to DRAM. O

- working set이 hundreds of megabytes 일 수 있음을
  언급하며 last level cache에 안맞음.

- Given the use of hardware multithreading to hide DRAM
  latency. the chip area used for cache in system
  processors is spent instead on computing resources
  and on the large number of registers to hold the state of
  the many threads of SIMD instructions.