Sample Sort on GPUs
Due: March 2, 2017

Implement the sample sort algorithm covered in the class using CUDA. Optimize your code with the possible CUDA optimizations - e.g., confining most of the access to the shared memory, reducing warp divergence, warp-based synchronization, coalesced access etc. Compare the execution time with the sequential CPU agorithm on the CPU for the largest data size involving integers that your implementation can accommodate on the GPU. Use the turing GPU node for this assignment. See Platform Notes -> First CUDA program.
Your assignments will be relatively marked based on the maximum problem size and the performance shown.

Extra points: Implement the OpenMP version on the CPU host and compare the GPU and the CPU code performance.
More extra points: Implement a hybrid version that uses both the CPU host and the GPU for sorting the elements. Show the performance.