GPU Programming 2017/18

Tutorial 2

In this tutorial we will look at thread parallelism in C++. Our example problem will be the computation of the norm of a vector a:

$$\|a\| = \sqrt{\sum_{i=1}^{n} a_i^2}.$$

1.) Download the skeleton code and generate the build system using `cmake`; in a sub-directory `./build` type `cmake ..` on the command line to generate the make file or IDE project.

2.) Allocate an integer array `a` for storing `n` elements of type `int` on the heap using `malloc()`. Ensure that the program correctly cleans up the memory.

3.) Initialize the memory so that all elements of `a` have the value 1.

4.) Generate `num_threads` threads that compute the norm in parallel (no additional global variables should be used). Devise an appropriate parallelization strategy for this.

   i.) Write a naïve implementation that does not consider potential race conditions.

   ii.) Write a refined version where race conditions are avoided.

5.) Re-write your code so that the number of threads can be passed as a command line argument to the program, i.e. no changes to the code are necessary as the number of threads changes.

6.) How does the performance of your code scale as a number of threads?

Please finish the implementation until next week (week of 1/11/2016).