CUDA: GPU basics

1 Preliminary

To use CUDA, add `module load cuda` in your `.bashrc` or `.bash_profile`. Also, if you have a `module load gcc` in there, replace it with `module load gcc/5.3.0`

To submit a CUDA job manually, use `-lnodes=1:ppn=7:gpus=1` in you `qsub` command. But `make bench_gpu` will take care of that.

2 Polynomial expansion

The problem is simple. It is to compute a polynomial function F of degree d:

$$F(x) = \sum_{i=0}^{d} a_i x^i$$

for an array of n values of x.

(The code for polynomial expansion on the CPU is provided.)

Question: Write a simple CUDA code that allocates and fill an array on the CPU and transfer it to the GPU. (Take array size as a parameter)

Question: Compute the polynomial expansion of each element of the array on the GPU. (Take block size and degree of the polynomial as a parameter.)

Question: Bring the results back on the CPU and confirm the GPU code is correct. You can test your code by running `make test` which will queue a job and return the results in `test_gpu.sh.o`***xyz*** and `test_gpu.sh.e`***xyz***

Hint: Remember to check explicitly for errors!

3 Measurements

Question: Measure the runtimes of the codes with `make bench_cpu` and `make bench_gpu` for polynomial function of degree 1, 10, 100, 1000, and for $n = 10^9$.

Question: Compare to the performance of the CPU implementation. You can compute a speedup table with `make table`. Which is faster in which configuration? Why do you think that is?