GPU Programming with CUDA

Instructor: Associate Professor Dan Negrut, NVIDIA CUDA Fellow
University of Wisconsin-Madison
http://sbel.wisc.edu

Day 1 – Wd, June 6

10:00 – 13:00 Lecture [CUDA intro]

- Introduction: Example use of GPU computing in Computer Aided Engineering
- Quick overview of trends in parallel computing (multi-core and GPU computing); Top500 list
- NVIDIA's CUDA intro: computation model and execution configuration
- CUDA memory allocation
- CUDA example: matrix multiplication

13:00-14:00 Lunch

14:00-17:00 Hands-on component

- Scaling a vector on the GPU
- Vector addition
- Dot product

Assignment (elective, due on Thursday morning):

- Parallel reduce operation

Day 2, Th June 7

10:00 – 13:00 Lecture [More advanced CUDA features]

- CUDA Memory model: registers and global, constant, texture, shared, local memories
- CUDA execution scheduling; thread divergence
- CUDA streams
- CUDA optimization rules of thumb

13:00-14:00 Lunch

14:00-17:00 Hands-on component

- Dot product, revisited – using shared memory to improve performance
- Matrix multiplication: Large, tiled matrix-matrix multiplication with and without shared memory

Assignment (elective, due on Friday morning)

- Matrix convolution

Day 3, Fr June 8

10:00 – 13:00 Lecture [Productivity tools]

- The CUDA **`thrust`** library
- CUDA profiling
- CUDA debugging
- The CUDA library landscape

13:00-14:00 Lunch

14:00-16:00 Hands-on component

- example of using the **`thrust`** library: reduction and prefix scan operations
- profiling of dot product using **`nvvp`**