4-day CUDA Course at Airbus Defence and Space

(29 September - 2 October 2015)

Applied Parallel Computing LLC is delivering GPU training courses since 2009. Several dozens of courses have been organized all over Europe, both for commercial and academic customers. We work in close partnership with NVIDIA, CUDA Centers of Excellence and Tesla Preferred Partners. In addition to trainings, our company provides GPU porting/optimization services and CUDA certification.

All corresponding presentations and code samples will be available to attendees from the beginning of each training day.

Hands-ons: All discussed topics will be accompanied with practical sessions, using NVIDIA CUDA 7.0 toolkit. Exercises will be conducted on the provided GPU-enabled laptops.

Day 1: Introduction to GPU Computing and CUDA

Morning (09:00-12:30)

09:00-10:30: lecture

- An overview of GPU performance in various applications
- Brief intercomparison of different types of accelerators
- Key programming principles to achieve high GPU performance

10:30-12:30: hands-on session

- Login into laptops
- Understanding GPU compute capabilities, *deviceQuery*
- Overview of CUDA applications development using Eclipse IDE
- Exercise 1: Compile & deploy poissonGL CUDA demo in Eclipse IDE

Afternoon (13:30-17:00)

13:30-15:00: lecture

- CUDA principles and CUDA implementation for C++
- Analogies between MPI+OpenMP and CUDA programming models
- The first CUDA program explained
- CUDA compute grid, examples
- Realistic CUDA application example (wave propagation code)
- Basic optimization techniques

15:00-17:00: hands-on session

- Exercise 2: Write & deploy a simple CUDA program
- Exercise 3: More control on CUDA compute grid
- Exercise 4: Write & deploy a meaningful image processing tool in CUDA

Day 2: GPU-enabled libraries and GPU memory hierarchy

Morning:

- CUBLAS, MAGMA, CUBLAS-XT, CUSPARSE, CUFFT, CURAND, CUSP
- AmgX – Krylov and mutigrid solvers
- **Hands-on:** solving Poisson equation with CUFFT
- **Hands-on:** solving linear system with AmgX library (*amgx_precond*)

Afternoon:

- GPU memory types
- Shared memory
- GPU caches hierarchy and mode switches
- Automatic texture cache (Kepler GK110)
- Unified virtual address space (UVA) in CUDA 6.0
- Streams and asynchronous data transfers
- **Hands-on:** “fill-in” exercise on reduction with and without shared memory
- **Hands-on:** getting additional performance using automatic texture cache

Day 3: Advanced CUDA, optmization & profiling

Morning:

- Dynamic parallelism
- Dynamic memory allocation in CUDA threads
- Compiling & linking relocatable device code
- CUDA C++ compiler pipeline, PTX assembler, SASS
- Understanding “-Xptxas -v” reports

Afternoon:

- PCI-E optimizations: streams, asynchronous data transfers
- An overview of Fermi, Kepler and Maxwell GPU architectures
- GPU optimizations: compute grid, coalescing, divergence, unrolling, vectorization, maxrregcount, aligning, floating-point constants
- Overview of *NVIDIA Visual Profiler*
- Overview of *nvprof* (command line profiler)
- Common practices of identifying performance hazards in GPU application using NVIDIA Visual Profiler
- **Hands-on:** profile and optimize the bilinear interpolation kernel

Day 4: Debugging & CUDA vs OpenCL

Morning:

- Principles and terminology
- GNU Debugger (*gdb*)
- CUDA-enabled GNU Debugger (*cuda-gdb*)
- GPU memory checker (*cuda-memcheck*)

- Debugging SASS without the source code
- **Hands-on:** live demonstration of *cuda-gdb* debugger on a sample application

Afternoon:

- Terminology of CUDA and OpenCL
- NVIDIA and AMD GPUs: architectural differences
- Execution workflow: offline compilation in CUDA, runtime compilation in CUDA (7.0) and in OpenCL
- **Hands-on:** CUDA and OpenCL intercomparison, by example of bilinear interpolation kernel