

Programming on the GPU with CUDA

Two-day course for researchers, PhD and MSc students

January 21-22, 2019

What is a GPU? Nowadays the Graphics Processing Unit (GPU) is a mainstream hardware component in high-performance computing. For affordable budgets anyone can harness supercomputer performance. However, realizing efficient parallelism combines three knowledge areas: firstly, on the architecture and compute capabilities of the GPUs; then, on special constructs for programming a GPU-equipped computer; finally, on the special algorithms for performing logical and mathematical operations in parallel.

What is CUDA? CUDA stands for Compute Unified Device Architecture. It is a software-development tool kit for programming on the GPUs maintained by the mainstream manufacturer Nvidia. CUDA provides language extensions for C, C++, FORTRAN, and Python as well as knowledge-specific libraries. A single source code is then able to instruct the CPU and GPU alike. Also, CUDA-extended codes keep pace closely with the rapid developments in the underlying technology.

Goals and prerequisites To guide you in this development niche, the [Delft Institute for Computer Science and Engineering](#) (DCSE) offers a 2-day course every quarter. We will explain basic principles and advanced topics on GPU programming with CUDA. You will apply these notions in our labroom with hands-on examples. After this course you will be able to get simple CUDA programs running on a GPU-equipped computer.

As prerequisite, a rudimentary understanding of programming languages like C++ or Java is ideal; that of FORTRAN or Python will be helpful too. Some interest in linear algebra and iterative solvers is a little advantage.

Instructors and schedule Prof Kees Vuik (KV), Mr Kees Lemmens (KL), Dr Matthias Möller (MM) of the Faculty of Electrical Engineering, Mathematics and Computer Science teach the course.

	Basic principles (day 1)	Advanced topics (day 2)
09:15–09:30	Arrival: refreshments	Arrival: refreshments
09:30–10:30	Parallel computing on GPUs KV	Paralution KV
10:45–11:30	GPUs: design and architecture KL	Lab 4: Shared memory, streams, atomics KL
11:45–12:30	Lab 1: CUDA introduction KL	Lab 5: Optimising code: inner products KL
12:45–13:30	Lunch	Lunch
13:30–14:45	Lab 2: Using CuBlas, CuFFT KL	Lab 6: Unified memory, dynamic parallelism KL
15:00–16:30	Lab 3: Debugging and profiling KL	Parallel solvers on GPUs MM
16:30–17:00	Lab 3: Debugging and profiling KL	Lab 7: Solvers on GPU MM

Location Delft University of Technology, Faculty of Mathematics & Computer Science, Van Mourik Broekmanweg 6, 2628 XE Delft. The classes are given in Building 28, ‘Penguin’ Laboratory Room (E0.380).

Costs and registration The course is free for DCSE members. TU Delft staff and students pay € 50/100 for attending one/two-day(s) respectively. The fee for other participants is € 200/350. Members of the SPHERIC research community receive a 20% discount on the full fee. Lunch, refreshments, lecture materials are included. Please sign up at <https://www.aanmelder.nl/gpucourse2019>. The maximum number of participants is 20.

More info The course’s splash page is <https://www.tudelft.nl/cse/education/courses/cuda-course/>. Please contact C.W.J.Lemmens@tudelft.nl for content or dcse@tudelft.nl for administration and logistics.

