

# NCT-300 Programming GPU Processors

Four day intensive training course covering all aspects of programming GPU processors using advanced methods and techniques

## ● Course Overview

This course covers concepts and approaches related to programming GPU processors using both CUDA and OpenCL. Extensive coverage of GPU hardware, memories, data transport and performance optimization enable the student to understand the fundamental aspects of GPU programming. In-depth, hands-on laboratories demonstrate how to apply common numerical methods to GPU processors using both the native APIs and open source numerical libraries. This course also covers methods of integrating the Intel TBB threading abstraction layer with GPU software APIs.

## ● Course Objectives

- Install GPU libraries and drivers, compile CUDA/OpenCL programs on Linux and Windows operating systems.
- Understand nVidia GPU hardware and the underlying technical concepts including SIMD processing and hardware threading architectures.
- Understand the different GPU programming APIs and their appropriate use with various applications.
- Understand single and double precision floating point calculations.
- Understand the difference between GPU memory types and the advantages and disadvantages of each.
- Effectively orchestrate the transport of data to and from GPU memory.
- Correctly implement two common types of numerical algorithms - Matrix Multiplication and Reduction.
- Coverage of performance optimization including the cudaprof profiling tool, loop unrolling, coalesced memory access, memory bandwidth estimations and occupancy.
- Introduction to and labs for commonly used open source computational libraries - CUBLAS, CUFFT and CUDPP.
- How to meld multicore processors and GPUs to take maximum advantage of modern platform performance.
- How to integrate the Threading Building Blocks threading abstraction layer with GPU code and migrate TBB primitives to the GPU.
- Discover how to take advantage of multiple GPUs in the same server.

## ● Benefits

- Teaches everything necessary to start developing high-performance GPU software on Linux or Windows platforms.
- Coverage of both CUDA and OpenCL including open source computational libraries such as CUBLAS, CUFFT and CUDPP.
- How to integrate multicore software development techniques with GPUs to increase performance.
- A comprehensive training workshop: This course offers an in-depth overview of fundamental concepts while offering advanced training and practical advice on GPU programming.
- Online training delivery platform combined with instructor led, hands-on laboratories provide in-depth instruction and increase students knowledge and skills.

- Coverage of cudagdb for CUDA debugging including the use of emulation mode with Valgrind.
- Extensive hands-on laboratories with code examples using both CUDA and OpenCL.

## ● Who should attend

Software architects, software developers, software team leaders and managers seeking to develop GPU software. Knowledge of computer architectures, intermediate C++ programming and software development experience are mandatory pre-requisites for this course.

## ● About nCore

nCore Design LLC develops high performance software and systems for multicore processors, GPUs and embedded systems. nCore has significant expertise on wide variety of computing platforms and specialized knowledge of applied software optimization techniques. nCore is a leading provider of hands-on, instructor-led training courses focused on programming multicore and GPU processors.

nCore is a privately held company with offices in the US and Japan. nCore works with local partners to deliver state-of-the-art solutions to multinational technology companies.

## Schedule - Day One

### Part One

- **Introduction**
- **Architecture** — History of Parallel/Vector Machines, GPU Evolution, nVidia Tesla Hardware Overview
- **API** — History, OpenGL Shader, HLSL/Cg, Sh/Brook, Current GPU API and Example, CUDA/Stream and Example, OpenCL and Example

### Part Two

- **CUDA** — Low Level API, High Level API, Differences
- **OpenCL** — API Overview
- **Lab: First Steps**
  - Installation
  - Compilation
  - Execution
  - Basic Timing

## Schedule - Day Two

### Part One

- **Memories** — Global/Shared
  - Local
  - Register
  - Constant
  - Texture
- **Memory Performance** — Coalesced access to memory
- **Lab: Timing memory access for each memory type**
  - Access with offset
  - Access with stride

### Part Two

- **Lab: Matrix Multiplication**
  - Naive Implementation: loop
  - Multithreaded Implementation
  - Using Shared Memory
  - Performance Comparison
- **CUBLAS** — Matrix Computation Library, Description
- **Lab: Examples of Use**
  - Performance Comparison
- **Accuracy** — Single/Double Precision Floating Point, Associativity
- **Lab: Precision Exploration Lab**

## Schedule - Day Three

### Part One

- **Lab: Reduction**
  - Naive Implementation
  - Optimization - Shared memory and coalesced access
- **Lab: Loop Unrolling**
  - Branch Optimizations

### Part Two

- **CUFFT** — Description and Examples
- **Lab: CUFFT Usage**
- **CUDPP: Data Parallel Primitives Library**
  - Parallel Reduction and Parallel Scan
  - Examples and performance comparison
  - Applications

## Schedule - Day Four

### Part One

- **Multicore and GPU** — Asynchronous API to GPU: Overlapping Computations
- **Lab: Asynchronous API**
- **TBB and CUDA**
- **Lab: Managing threads to retrieve GPU data**
- **Lab: Migrating TBB primitives to CUDA/OpenCL**
- **Lab: Using TBB pipeline with CUDA/OpenCL**
- **Lab: Thread Pool using GPU resources**
- **MultiGPU** — Using more than one GPU
- **Lab: MultiGPU Exploration Lab**

### Part Two

- **Debugging**
  - cudagdb
  - Emulation Mode
  - Valgrind Example
- **Lab: Debugging**
- **Performance**
  - cudaprof
  - Occupancy
  - **Lab: Occupancy Evaluation and Improvement**
  - Memory Bandwidth
  - Theoretical Estimation
- **Best Practices**
- **Resources** — Key resources and further Study

### Course Registration



<http://www.ncoredesign.com/training/>

**Length:** 4 Days

**Cost:** \$3495

#### Registration Methods

**Online:** <http://www.ncoredesign.com/training/>

**By E-mail:** [training@ncoredesign.com](mailto:training@ncoredesign.com)

**By Fax:** 971-228-0707

