

NCT-100 Programming Multicore Processors

Three-day intensive training course covering all aspects of programming multicore processors using advanced methods and techniques

Course Overview

This course covers concepts and approaches for programming multicore processors in C/C++. From recognizing parallelism opportunities to designing multithreaded algorithms, this course teaches students how to deal with the necessary aspects of multithreading, synchronization, multi-core processor caches, shared memory, and specialized methods for multithreaded programming using modern methods, such as OpenMP and Intel Threading Building Blocks.

Course Objectives

- Review theoretical background covering multicore processor architecture, concurrent programming, and parallel programming concepts and considerations.
- Cover critical concepts such as implicit and explicit parallelism, atomicity, synchronization, shared memory, cache coherency, Amdahl's Law, Flynn's processor classifications, and Little's law in detail.
- Recognize the best parallelism opportunities and explain the advantages of using threads to obtain concurrency using various analysis techniques, compositional approaches, and parallel design patterns.
- Learn how to avoid synchronization pitfalls such as starvation, deadlock, live lock, and data races.
- Learn how to implement and tune parallel algorithms.
- Define and use different synchronization methods effectively, including mutexes/critical sections and condition variables.
- Explain operating system interactions and the relationship between shared memory and threads.
- Explain what aspects of the operating system affect programming, how to deal with shared memory effectively, CPU selection, CPU-specific binding of threads, thread specific data, and kernel-level scheduling.
- Understand and use threads with specific technologies and programming methods, such as the Windows API,

Benefits

- **A comprehensive training workshop:** This course offers an in-depth overview of fundamental concepts, while offering advanced training and practical advice on C/C++ programming of multicore processors using modern methods.
- **Gain critical insights on how to improve your software's performance:** This course is designed to give you key skills using specialized tools to help you to correctly architect, design, and develop efficient parallel applications for multicore processors.
- **Additional hands-on learning:** This course provides laboratory sessions in writing multithreaded programs and exercises on practical parallelization of legacy software. It also includes walk-through laboratory exercises designed to increase your understanding of multithreading.

POSIX pthreads, Intel TBB, and OpenMP using C/C++ and the Intel Compiler.

- Gain hands-on experience with the Intel Compiler to build and run multithreaded programs during the laboratories.
- Learn best practices to deal with MT-unsafe libraries and how to write new thread-safe libraries.

Who should attend

Software architects, software developers, software team leaders and managers seeking to understand and implement efficient software running on multicore processors. Knowledge of the C++ programming language and C++ software development experience is a pre-requisite for this course.

About nCore

nCore HPC is a global provider of professional services and systems focused on high performance, low latency and scalability in embedded computing. nCore delivers state-of-the-art solutions to government agencies, high-technology organizations, defense, research, biomedical and financial companies.

nCore is a working group member of the Multicore Association.

Course Registration

Length: 3 Days Cost: \$2495

By E-mail: training@ncorehpc.com