

Архитектура и Программирование Современных Многоядерных Процессоров (Компактный курс лекций с практикой , 14-16 часов, письменный экзамен 90 минут)

Проф. Сергей Горлач (Университет Мюнстер, Германия)/ Prof.Dr. Sergei Gorlatch (WWU Münster)

В предлагаемом курсе студенты изучат архитектуру и организацию современных высокопроизводительных процессоров, основным свойством которых является многоядерность и, как следствие - высокая степень параллелизма в программировании этих процессоров.

Студенты овладеют практическими навыками по вычислению пиковой производительности таких процессоров, а также производительности типичных программ и приложений.

В курсе будут даны теоретические основы и практические навыки по программированию многоядерных процессоров с использованием популярных современных подходов: PThreads и OpenCL. Студенты научатся разрабатывать программы с помощью этих подходов, выявлять главные источники ошибок в программах и устранять их.

Особое внимание будет уделено практическим методикам оптимизации многоядерных программ с использованием разнообразных алгоритмических и программных приёмов.

Пререквизиты для участия в курсе со стороны студентов: базовые знания в языке программирования C, базовое понимание взаимодействия архитектуры, операционной системы и компилятора при выполнении программ в компьютерах. Для практического опробования материала (по желанию) понадобится установить Pthreads и/или OpenCL.

Материалы курса (слайды) - на английском языке, будут предоставлены заранее. Курс может читаться, по желанию, на русском или английском языке. Письменный экзамен - на английском языке, ответы принимаются на любом из языков: английский, немецкий, русский.

Курс является сокращенной, адаптированной и переведенной на английский версию курса, читаемого в Университете Мюнстера (Германия) на уровне Бакалавра или Магистра.

Программа курса (на английском):

Architectures and Programming of Modern Many-Core Processors (Course Program):

Introduction: Motivation for parallelism, History, New Challenges with multi- and many-cores

Theory: Architectures of CPU and GPU – Levels of parallelism, Comparison of different vendors and models, Numbers and characteristics of typical contemporary hardware

Practice: Calculating peak performance, comparing machine programs for different architectures

Theory: Thread programming for multi-core CPUs, PThreads

Practice: Hyperthreading and cache optimization

Theory: OpenCL: Motivation, Concepts and principles

Practice: Writing and executing OpenCL programs

Theory: OpenCL: Execution on CPU vs. GPU

Theory: Memory hierarchy in GPU, optimization for memory in OpenCL programs

Theory: Advanced optimizations in OpenCL: Warps etc.

Practice: optimizations for matrix multiplication, Sobel filter, Mandelbrot set

Optional material: Alternative programming approaches: CUDA, OpenMP, MPI, etc.

Writing exam: 90 min.