Summary
This course provides insight into a broad variety of High Performance Computing (HPC) concepts and the majority of modern HPC architectures. Moreover, the student will learn to have a feeling about what architectures are suited for several types of algorithms.

Content
HPC overview:
• Today's HPC: Beowulf-style clusters, massively parallel architectures, hybrid computing, accelerators
• HPC history and background
• HPC benchmarks explained
• Multicore systems
• Scaling

Writing HPC code:
• Shared memory parallelism with OpenMP
• Distributed memory parallelism with MPI
• Hybrid programming with OpenMP and MPI
• GPGPU primer
• Profiling

Keywords
HPC, Parallelization, MPI, GPU

Learning Prerequisites
Required courses
• Analysis, bachelor level
• Numerical analysis for engineers
• Matrix algebra
• Programming concepts in scientific computing
Learning Outcomes
By the end of the course, the student must be able to:
• Classify the types of HPC architecture
• Identify codes suited for parallelizing
• Apply the most common parallelization techniques
• Implement algorithms in parallel
• Investigate the performances of parallel code
• Argue about the differences in performances between theory and practice
• Optimize the usage of hardware and software resources depending on the type of algorithm to parallelize

Transversal skills
• Set objectives and design an action plan to reach those objectives.
• Communicate effectively with professionals from other disciplines.
• Access and evaluate appropriate sources of information.
• Write a scientific or technical report.

Teaching methods
Lectures, exercises, project work

Expected student activities
Attendance at lectures, completing exercises, writing a project

Assessment methods
Oral defense of project work

Supervision
Office hours Yes
Assistants Yes
Forum Yes