

## 586SM - FOUNDATIONS OF HIGH PERFORMANCE COMPUTING

### Aims

On successful completion of this course students should be able to understand all the basic concepts of modern High Performance Computing platform. They should be able to use them and write parallel program using different paradigm as well.

Knowledge and understanding: the student must have acquired the necessary knowledge to understand in details all the elements of a parallel architecture and the paradigm and the techniques to program them at best. They have a clear idea about the basic HPC concepts.

Applied knowledge and understanding: the student must have learned how to use a HPC platform, how to submit serial and parallel task, how to perform a micro benchmark of the system, how to write, compile and execute a parallel code by means of message passing and shared memory paradigm. They should be able to write short report on scalability and performance analysis.

Making judgments: the student must be able to correctly identify (if any) issues/problems of a parallel program and/or HPC platform by means of the results of a benchmark analysis and/or the performance evaluation.

Communication skills: the student must be able to communicate effectively the results of a benchmark analysis of an HPC platform and/or of a parallel program. They should be able to properly communicate weaknesses and problems to the HPC sys. administrator and HPC management.

Learning skills: The students must demonstrate to have the basic skill set needed to understand the key-features of a modern parallel systems for HPC, including the ability to perform basic benchmarks and produce clear reports on the performance results. Moreover, students must have the technical knowledge to understand the main programming paradigms commonly used in scientific programming for HPC.

### Teaching Format

The course will comprise different approaches as specified below and will be delivered by more than one teacher. There will be also invited experts in the field that will cover some specific topics.

Students are supposed to take active part in the course: besides teachers' lectures, short assignments will be given to students that are expected to accomplish the tasks during the class. Solutions and errors will then be discussed together.

Students are moreover requested to use several software (compiler/libraries): this will require to download, install and run it for the first time. We will give a short practical introduction to it.

### Assessment

The exam will consist in a written exam, and a short software project to be completed individually by the student and delivered before the exam. The software project implies the use of a git repository where the code should be committed within a certain deadline