

Course syllabus

Course content. This course is aimed at giving a first introduction to classical topics in numerical analysis, including floating point arithmetics, numerical integration, numerical solution of linear and nonlinear equations, iterative methods for eigenvalue problems, numerical schemes for ordinary differential equations, and optimization.

Prerequisites. The course assumes a basic knowledge of linear algebra and calculus. Prior programming experience in Julia, Python or a similar language is desirable but not required.

Study goals. After the course, the students will be familiar with the key concepts of *stability*, *convergence* and *computational complexity* in the context of numerical algorithms. They will have gained a broad understanding of the classical numerical methods available for performing fundamental computational tasks, and be able to produce efficient computer implementations of these methods.

Education method. Although theoretical questions are touched on concerning, for example, the convergence and stability of the methods examined, the emphasis in this course is placed on practical aspects of scientific computing: efficient computer implementation, testing, debugging, validation and visualization. The course comprises six sessions, each with a duration of 2h45 and the following structure:

- **Before the session**, the student prepares a theoretical exercise or reads a part of the lecture notes corresponding to the subject of the session.
- **(30 min)** Each session starts with the presentation and analysis of one or two simple methods. We study, for example, the power method for approximating the eigenvalues of a matrix, and the explicit Euler method for solving ordinary differential equations.
- **(90 min)** The main part of the session is dedicated to practical exercises in small groups of 4 or 5 students. The exercises blend theoretical questions and practical computer implementation tasks.
- **(30 min)** At the end of each session, an overview of the other numerical methods popular within the particular field covered in the session is presented, with the aim of providing

students with the necessary information to choose the appropriate method in different contexts of practical interest. These lecture contain additional details, including convergence proofs, for interested students.

Assessment. A problem set is handed out after each session. Students must submit three out of the six problem sets for evaluation. The final mark is calculated from these and a short oral discussion with the teacher. The rule followed for marking computer programs returned by the students is the following:

- The mark for a code that contains a syntax error is 0/20;
- The mark for a code that runs but does not work is at most 14/20;
- The mark for a code that works is at least 16/20;
- For a code that works, the precise mark is decided based on efficiency and style.

Literature and study material. A comprehensive reference for this course is the following textbook: A. QUARTERONI, R. SACCO, and F. SALERI. *Numerical mathematics*, volume **37** of *Texts in Applied Mathematics*. Springer-Verlag, Berlin, 2007. Other pointers to the literature will be given within each chapter.