COURSE OF STUDY THREE-YEAR BACHELOR PROGRAMME IN MATHEMATICS

ACADEMIC YEAR 2023-2024

ACADEMIC SUBJECT NUMERICAL CALCULUS 2

General information	
Programme year	Third
Term	Second semester (February 26, 2024 – May 31, 2024)
European Credit Transfer and Accumulation System credits (ECTS)	7
SSD	MAT/08 – Numerical Analysis
Language	Italian
Mode of attendance	Not mandatory

Lecturers		
Name and surname	Felice Iavernaro (instructor of record)	Giuseppe Vacca
E-mail	felice.iavernaro@uniba.it	giuseppe.vacca@uniba.it
Telephone	+39 080 544 2703	+39 080 544 2645
Department and office	Department of Mathematics room 2 fourth floor	Department of Mathematics room 12 second floor
Virtual meeting room		
Web page	https://www.dm.uniba.it/it/members/iavernaro	https://www.dm.uniba.it/it/members/vacca
Office hours		

Work schedule				
	Total	Lectures	Hands-on learning (recitations/laboratories)	Self-study
Hours	175	40	30	105
ECTS credits	7	5	2	

Learning objectives	
	Acquiring knowledge about numerical methods and programming techniques in the context of interpolation, data fitting, numerical integration, numerical solution of ordinary differential equations.

Course prerequisites	
	The knowledge acquired in the course “Numerical Claculus 1”, programming in Matlab, classical analysis of one and several variables, fundamental linear algebra.

Syllabus	
Course contents	1. INTERPOLATION The polynomial interpolation problem: existence and uniqueness theorem. Undetermined coefficient method: computational cost and stability issues. Lagrange formula and barycentric formulations. Remainder term in the polynomial interpolation. Convergence properties. Stability properties: Lebesgue function and Lebesgue constant. Newton divided differences

	interpolation formula: Newton basis and divided differences. Newton-Gregory forward difference formula. Runge's phenomenon. Chebyshev nodes and their optimality property. Chebychev-Lobatto nodes and related barycentric formula. Convergence results for differentiable and analytic functions. Hermite interpolation: extensions of Lagrange and Newton interpolation formulae, remainder theorem and computational properties. Linear splines: construction of a basis, stability, remainder and convergence properties. Cubic spline interpolation: extra boundary conditions, construction and implementation; hints about convergence and minimal curvature property. Generalized polynomials and Haar condition. Trigonometric interpolation: DFT and IDFT algorithms, computational complexity and implementation; some applications to the signal processing. 2. APPROXIMATION Overdetermined linear systems. Least squares approximation: problem definition, existence and uniqueness theorem. Least Squares fitting-polynomial. Linear regression line. Coefficient of determination. QR factorization of a rectangular matrix by means of Householder elementary transformations. Use of QR factorization for solving least squares problems. Singular Value Decomposition: existence result and fundamental properties. Solution of the least squares problem through the SVD decomposition. Pseudoinverse and its properties. Conditioning of the least squares problem. Bidiagonalization of a matrix by means of Householder elementary transformations. Algorithm for the SVD computation by means of Givens rotations. Some applications of the SVD: computation of the rank and the 2-norm of a matrix; best low-rank approximation of a matrix; regularizing an ill-conditioned linear system; digital image compression; latent semantic analysis; principal component analysis. Least squares problem in L2([a,b]): problem definition, use of an orthogonal basis, Bessel's inequality and Parseval's identity. Error estimation and implementation aspects. Orthogonal polynomials: Gram-Schmidt ortogonalization process, three-term recurrence relation, property of the roots. Rodrigues' formula and study of the family of Legendre polynomials, first and second kind Chebyshev polynomials, Laguerre and Hermite polynomials. Truncated Fourier series. Padé rational approximation. 3. NUMERICAL INTEGRATION Interpolation quadrature formulae, degree of precision and error analysis. Rectangle, trapezoidal and Simpson rules: definition and error analysis. Newton-Cotés formulae. Conditioning of the problem and stability of an interpolation quadrature formula. Composite trapezoidal and Simpson's rules, error analysis. Recursive algorithm with automatic step control based on the trapezoidal Simpson's rules. Higher-order formulae, weights positivity and convergence properties. Gauss-Legendre, Radau and Lobatto formulae. Hints on adaptive integration techniques. 4. NUMERICAL METHODS FOR INITIAL-VALUE PROBLEMS One-step methods: definition, consistency and convergence analysis. Explicit and implicit Euler methods, implicit midpoint and trapezoidal methods. Collocation methods: definition and link to Runge-Kutta methods. Existence and uniqueness of the collocation polynomial. Order analysis. Gauss, Radau, Lobatto collocation methods.

	5. PROGRAMMING ENVIRONMENT FOR SCIENTIFIC COMPUTING The implementation of the algorithms introduced in the course will be carried out in Matlab, which is a scientific computing environment equipped with a number of built-in functions and programming instructions. A particular attention will be paid to the study of the behavior of the solutions of a given problem in finite arithmetic.
Reference books	• L.N. Trefethen, Approximation Theory and Approximation Practice SIAM, 2013. • G.H. Golub, C.F. Van Loan, Matrix computations. Fourth edition. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 2013. • R. Bevilacqua, D. Bini, M. Capovani, O. Menchi, Metodi Numerici, Zanichelli, Bologna, 1992. • E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I. Nonstiff Problems. Springer Series in Comput. Mathematics, Vol. 8, Springer- Verlag 1987, Second revised edition 1993. • D. Bini, M. Capovani, O. Menchi, Metodi numerici per l'algebra lineare, Zanichelli. • K.E. Atkinson, An introduction to numerical analysis, Wiley, 1989 • D.M. Young, R.T. Gregory, A survey of numerical mathematics, Vol. I, Dover, 1988
Additional course materials	Handouts, notes and Matlab codes will be made available through an e-learning platform. Log-in information will be provided during the course starting days.
Repository	https://elearning-mat.hosting.uniba.it

Expected learning outcomes	
Knowledge and understanding	• Acquiring a knowledge about the most important numerical methods able to solve mathematical real-world problems, with particular reference to data-fitting, numerical integration and initial value problems. • Understanding and being able to explain issues related to the use of a computer for solving the above mentioned mathematical problems.
Applying knowledge and understanding	• The solution of mathematical problems by means of suitable algorithms enjoying good stability properties and low cost implementation. • Encoding and testing numerical algorithms and consistently interpreting computer results.
Soft skills	*Making judgements*: Being able to detect a proper numerical method for solving a given mathematical problem among those analyzed during the lectures.
	Communication skills: Being able to provide rigorous definitions of the analyzed mathematical problems and to discuss the related numerical methods, outlining their most prominent features.
	Learning skills: Capability of studying and solving problems similar, but not necessarily equivalent, to those faced during the teaching activities.

Teaching methods	
	Lectures and exercise sessions. Exercise sessions.

Assessment	
Assessment methods	The exam consists of an oral test which includes a discussion of the Matlab

	codes prepared by the student. Before taking the exam, the student should prepare a pdf report in which he presents the numerical experiences and the results obtained. The file should be sent to the teachers via email a few days before the exam.
Evaluation criteria	• *Knowledge and understanding*: Identification of the fundamental properties of the methods, with particular reference to the hypothesis of applicability and computational efficiency. Ability to compare methods that solve the same problem. • *Applying knowledge and understanding*: Discussion of the codes and examples performed; correct interpretation of the results obtained. • *Making judgement*: : Collecting and processing data, and interpretation of the results obtained. • *Communication skills*: Clarity, also in terms of formalism, in the description and coding of the numerical methods, as well as ability to effectively present the numerical tests carried out. • *Learning skills*: Numerical implementation and discussion of application problems more involved than those presented during the lectures.
Grading policy	The final grade takes into account, both the theoretical discussion of the methods studied and their correct implementation on the computer. The quality of the examples produced and the ability to illustrate them are also considered. Assessment is based on the achievement of the plssnned learning objectives. For a high evaluation, a thorough knowledge of the topics is required, as well as an adequate capacity for argumentation and exposition.

Further information	