University of Gdańsk

Subject card

Subject name and code	Foundations of Computational Mathematics, PG_00161072						
Field of study	Mathematical Modeling and Data Analysis						
Date of commencement of studies	October 2023		Academic year of realisation of subject		2025/2026		
Education level	Bachelor's studies		Subject group				
Mode of study	full-time studies		Mode of delivery		at the university		
Year of study	3		Language of instruction		English		
Semester of study	5		ECTS credits		6.0		
Learning profile	academic		Assessment form		exam		
Conducting unit							
Name and surname of lecturer (lecturers)	Subject supervisor		dr hab. Karolina Kropielnicka				
	Teachers		dr Paul Bergold dr hab. Karolina Kropielnicka				
Lesson types	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	30.0	0.0	0.0	60
	E-learning hours included: 0.0						
Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan		Participation in consultation hours	Self-study		SUM
	Number of study hours	60		6.0	50.0		116
Subject objectives	A gentle introduction to mathematical computations, where students will become familiar with fundamental numerical tools and concepts. The course covers topics such as problem conditioning, algorithmic stability, basic methods of interpolation, numerical integration (quadrature), and selected topics in numerical linear algebra. Emphasis is placed on developing an intuitive understanding of numerical methods and their practical applications.						

Learning outcomes

Course outcome	Subject outcome	Method of verification
[MMiADL3_K02] is ready to precisely formulate questions to deepen his/her own understanding of a given topic or to find missing elements of reasoning	The student understands fundamental concepts and methods in computational mathematics, including conditioning, stability, interpolation, quadrature, and basic numerical methods for differential equations.	[SK4] test/exam - oral or written
[MMiADL3_W09] knows and understands the basics of computational and programming techniques supporting mathematician's work and understands their limitations	The student can critically assess the quality of numerical results (e.g. in terms of rounding errors, convergence, and stability).	[SW4] test/exam - oral or written
[MMiADL3_U10] is able to recognise problems, including practical issues, that can be solved algorithmically; can make a specification of such a problem	The student can select and apply appropriate numerical methods to solve basic mathematical problems, particularly ordinary differential equations.	[SU4] test/exam - oral or written
[MMiADL3_U11] knows how to arrange and analyse an algorithm in accordance with the specification and save it in the selected programming language	The student is aware of the importance of numerical approximation and the limitations of computer-based computations in solving mathematical problems.	[SU4] test/exam - oral or written

Subject contents	1. **Introduction to Computational Mathematics:** Motivation and applications; when numerical methods are needed. The role of computers in solving mathematical problems. 2. **Number Representation and Computational Errors:** Floating-point arithmetic; absolute and relative errors. Error propagation and its effect on result accuracy. 3. **Conditioning and Stability:** Problem conditioning vs. algorithmic stability. Examples of well- and ill-conditioned problems. 4. **Interpolation and Approximation:** Polynomial interpolation (Lagrange, Newton). Runges phenomenon and stability issues. Introduction to function approximation. 5. **Numerical Quadrature (Integration):** Newton-Cotes rules (rectangle, trapezoidal, Simpsons). Accuracy and stability of integration methods. 6. **Basics of Numerical Linear Algebra:** Solving linear systems: Gaussian elimination, LU decomposition. Numerical errors and solution stability. 7. **Numerical Methods for Ordinary Differential Equations (ODEs):** Eulers method, Runge-Kutta methods (RK2, RK4). Application to simple evolutionary equations. Solution visualization. 8. **Practical Applications and Project Work:** Implementation in Python (NumPy, Matplotlib). Final project: solving and analyzing a selected mathematical problem using learned methods.
Prerequisites and co-requisites	**Prerequisites:** • Basic knowledge of calculus and linear algebra (typical for first-year university level). • Introductory skills in Python programming (working with variables, control structures, loops, functions). • Basic communication skills in English, sufficient for following the lectures and materials. **Recommended requirements:** • Interest in the application of mathematics in computational science or engineering. • Openness to working in a multilingual environment and willingness to use English-language resources. • Readiness to engage in small projects and independently solve programming and mathematical tasks.
Assessment methods and criteria	

Subject passing criteria	Passing threshold	Percentage of the final grade
written exam	51.0%	100.0%

Recommended reading	Basic literature	[SM] Abner J. Salgado, Steven M. Wise, Classical Numerical Analysis: A Comprehensive Course, Cambridge University Press, 2022 [DR] P. Davis, P. Rabinowitz, Methods of numerical integration (2nd ed.), Dover Publications, 2007 [D] J. Demmel, Applied Numerical Linear Algebra, SIAM 1997 [H] N. Higham, Accuracy and Stability of Numerical Algorithms, SIAM, 1996 [I] A. Iserles, A First Course in the Numerical Analysis of Differential Equations (2nd ed.), CUP, 2009. [KU] A. Krommer, C. Ueberhuber, Computational integration, SIAM, 1998 [S1] G. Stewart, Afternotes on numerical analysis, SIAM, 1996 [S2] G. Stewart, Afternotes goes to graduate school, SIAM, 1998
	Supplementary literature	[C] R. Caflisch, Monte Carlo and quasi-Monte Carlo methods, *Acta Numerica* 7, 1149, 1998 [Cheb] G. Wright, M. Javed, H. Montanelli, N. Trefethen, Extension of Chebfun to periodic functions, SIAM J. Sci. Comput. 37(5), 554573, 2015 [O] M. Overton, Numerical Computing with IEEE Floating Point Arithmetic, SIAM, 2001 [GW] G. Golub, J. Welsch, Calculation of Gauss quadrature rules, *Math. Comp.* 23(106), 221230, 1969 [T] L. N. Trefethen, Approximation theory and approximation practice, SIAM, 2013 [TB] L. N. Trefethen, D. Bau, Numerical Linear Algebra, SIAM, 1997
	eResources addresses	
Example issues/ example questions/ tasks being completed	• How does problem conditioning affect the accuracy of numerical computations? • Numerically compute the integral of a function using the trapezoidal and Simpsons rules, and compare results. • Solve an ordinary differential equation using Eulers method and the 4th order Runge-Kutta method; visualize the solutions. • Implement and test the Gaussian elimination algorithm for solving a system of linear equations.	
Work placement	Not applicable	