University of Gdańsk

# Subject card

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Subject name and code | Computational Mathematics of Evolutionary Equations, PG_00171117 | | | | | | |
| Field of study | Mathematical Modeling and Data Analysis | | | | | | |
| Date of commencement of studies | October 2024 | | 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 | 2 | | Language of instruction | | English | | |
| Semester of study | 4 | | 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 | | | | | | |
| 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 | | 10.0 | 80.0 | | 150 |
| Subject objectives | The aim of the course is to introduce students to numerical methods for solving evolutionary equations, i.e., equations that describe phenomena evolving in time and space, commonly arising in physics, biology, chemistry, and engineering. The course covers spatial and temporal discretisation techniques (e.g., finite difference, finite element, and spectral methods) and investigates the numerical algorithms used to solve such models. Special emphasis is placed on understanding key algorithmic properties such as accuracy, stability, efficiency, and robustness. | | | | | | |

| Learning outcomes | Course outcome | Subject outcome | Method of verification |
|---|---|---|---|
| | [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 able to develop and analyse a numerical algorithm in accordance with the specification of a given mathematical problem and implement it in a selected programming language (e.g. Python). | [SU1] oral statement/conversation/ discussion<br>[SU8] observation of student's independent or team work |
| | [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 is able to identify mathematical and practical problems that can be solved algorithmically, and can formulate their specification in a way suitable for computational implementation. | [SU1] oral statement/conversation/ discussion |
| | [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 is ready to formulate precise questions related to numerical and computational problems in order to deepen their understanding and identify gaps in mathematical or algorithmic reasoning. | [SK1] oral statement/conversation/ discussion<br>[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 understands the basic computational techniques and programming tools (particularly in Python) that support solving mathematical problems, and is aware of their limitations and potential sources of numerical error. | [SW4] test/exam - oral or written |

| | |
|---|---|
| Subject contents | Introduction to Evolutionary Equations:<br>• Examples from wave, diffusion, and Schrödinger-type problems.<br>Foundations of Computational Mathematics for Evolution Equations:<br>• Space-time discretisation: grid concepts, time/space steps.<br>• Algorithmic quality criteria: accuracy, stability, convergence, efficiency.<br>Time Discretisation:<br>• Explicit and implicit methods (Euler, Crank-Nicolson, Runge-Kutta).<br>• Stability and stiffness (A-stability, L-stability).<br>Space Discretisation:<br>• Finite difference methods (FDM).<br>• Finite element methods (FEM).<br>• Introduction to spectral methods and applications.<br>Solving Evolutionary Equations:<br>• Coupling space and time discretisation.<br>• Schemes for parabolic, hyperbolic, and Schrödinger-type equations.<br>Numerical Analysis of Algorithms:<br>• Error sources: truncation, rounding.<br>• CFL condition and its implications.<br>• Conservation of physical quantities (e.g. energy) in numerical schemes.<br>Implementation and Visualisation:<br>• Python implementation (NumPy, Matplotlib, SciPy).<br>• Visualising solutions time evolution, spatial profiles.<br>Selected Topics and Student Projects:<br>• Applications to realistic problems (e.g. wave propagation, heat transfer).<br>• Projects using real-world physical or biological models. |

<table>
<tr><td>Prerequisites<br>and co-requisites</td><td>Prerequisites:<br><ul><li>Basic knowledge of calculus (derivatives, integrals, elementary differential equations).</li><li>Familiarity with linear algebra (systems of linear equations, matrices, vector spaces).</li><li>Basic programming skills in Python (variables, loops, functions, use of numerical libraries like NumPy).</li><li>Ability to communicate in English at a level sufficient for following lectures and participating in exercises.</li></ul>Additional recommended preparation:<br><ul><li>Completion of the Foundations of Computational Mathematics course (not required but helpful).</li><li>Familiarity with basic numerical methods (e.g. Eulers method, interpolation, numerical integration).</li><li>Interest in applications of mathematics to natural sciences and engineering.</li></ul></td></tr>
<tr><td>Assessment methods<br>and criteria</td><td><table><tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr><tr><td>written exam</td><td>51.0%</td><td>100.0%</td></tr></table></td></tr>
</table>

<table>
<tr><td rowspan="3">Recommended reading</td><td>Basic literature</td><td>1. Peter Deuflhard, Folkmar Bornemann, Scientific computing with ordinary differential equations, Springer, 2002.<br>2. Peter Deuflhard, Martin Weiser, Adaptive numerical solution of pdes, DeGruyter, 2012.<br>3. Ernst Hairer, Gerhard Wanner, Solving ordinary differential equations II, 2nd edition, Springer, 1996.<br>4. A. Iserles, A First Course in the Numerical Analysis of Differential Equations, CUP, 2009.</td></tr>
<tr><td>Supplementary literature</td><td>1. Ben Leimkuhler, Sebastian Reich, Simulating Hamiltonian dynamics, CUP, 2004.<br>2. C. Lubich, From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis, EMS, 2008.</td></tr>
<tr><td>eResources addresses</td><td></td></tr>
<tr><td>Example issues/<br>example questions/<br>tasks being completed</td><td colspan="2"><ul><li>Differences between explicit and implicit time discretisation methods.</li><li>Numerical stability the concept of A-stability and its importance.</li><li>Space discretisation: comparison of finite difference, finite element, and spectral methods.</li></ul></td></tr>
<tr><td>Work placement</td><td colspan="2">Not applicable</td></tr>
</table>