COURSE DESCRIPTION CARD - SYLLABUS

Course name
Numerical methods [S1MNT1>MN]

Course

Field of study
Mathematics of Modern Technologies

Year/Semester
2/3

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes
45

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

6,00

Coordinators

dr inż. Karol Gajda
karol.gajda@put.poznan.pl

Lecturers

Prerequisites

The student has a knowledge of mathematics (range: linear algebra, differential and integral calculus, initial value problems for ordinary differential equations), and computer science (for programming in high level language). The student is able to solve math problems analytically within the range specified above. The student is able to implement a computer program. The student is aware of the level of his knowledge. The student is aware of deepening and expanding knowledge.

Course objective

Understanding the basic numerical methods. Applying them to solve mathematical problems and simple engineering problems. Supporting mathematical and engineering calculations with appropriate IT tools. Verifi- cation of obtained solutions.

Course-related learning outcomes

Knowledge:
• the student has knowledge about the use of mathematical methods and tools in the field of numerical methods;
• the student has theoretically founded knowledge of numerical methods;

• the student knows at least one software package or programming language;
• [K_W01(P6S_WG), K_W07(P6S_WG)].

Skills:
• the student is able to use knowledge in higher mathematics;
• the student can use numerical tools and methods to solve simple engineering problems;
• the student can construct the algorithm of solving a simple engineering task and implement it and test it in the chosen development environment;
• the student is able to operate the devices in accordance with general requirements and knows how to apply the principles of health and safety at work in a computer laboratory;
• the student can use the knowledge and methods and tools to solve typical engineering tasks;
• thestudentknowshowtouseaforeignlanguagetotheextentthatitispossibletouseEnglish-language software;
•
[K_U01(P6S_UW),K_U03(P6S_UW),K_U04(P6S_UW),K_U05(P6S_UW),K_U06(P6S_UW),K_U11(P6S_UW

K_U12(P6S_UW), K_U15(P6S_UK)].

Social competences:
• the student is aware of the level of his knowledge;
• the student is aware of deepening and expanding knowledge to solve technical problems;
• the student is able to think and act in a creative way, is aware of the responsibility for the effects of the work of the team, as well as its individual participants; [K_K01(P6S_KK), K_K02(P6S_KK), K_K03(P6S_KO)].

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lectures:
• assessment of knowledge and skills demonstrated on the exam;
• control of perception during lectures;
Laboratory classes:
• assessment of skills related to the implementation of project tasks;
• assessment of student preparation for laboratory classes;
• assessment of skills related to the implementation of laboratory tasks;
• assessment of teamwork skills.

Programme content

Update: 01.06.2023r.
Lectures& Laboratory classes:
• floating point arithmetic, numerical errors;
• stability and accuracy of algorithms;
• numerical solutions of nonlinear equations and systems of equations (selected methods);
• polynomial interpolation;
• Taylor series;
• Newton-Cotes quadrature rules;
• numerical differentiation;
• initial-value problems for first-order ordinary differential equations (selected one-step methods of Runge-Kutta type).

Course topics

The lecture program covers the following topics:
1. Floating-point arithmetic.
Real number – various forms of notation.
Factoring numbers.
Converting numbers between decimal and binary systems and related problems.
Rounding and error measures.
Floating-point representations of real numbers.

Real numbers and machine numbers.
Machine accuracy.
Arithmetic operations with floating-point numbers.
Characteristics of floating-point arithmetic on a selected example.
Numerical errors.
Stable and unstable algorithms.
Task conditioning.

2. Numerical differentiation: Taylor's formula.
Discretization of areas. Characteristics of mesh methods.
Derivation of method formulas.
Order of convergence of O(*) methods.
Estimating errors.

3. Numerical methods for solving initial value problems for
ordinary differential equations of the first order, selected one-step methods of the Runge-Kutta type.
Graphical interpretation.
Local and global error. Total solution error.

4. Numerical solution of nonlinear equations of functions of one variable.
Selected iterative methods: The method of tangents (Newton-Raphson), bisection, secants and regula-falsi.
Graphical interpretation of methods.
Derivation of method formulas.
Conditions for convergence of methods. Local and global convergence.
Examples of method divergence.
Computational complexity of methods.
Termination conditions.
Verification of the correctness of the solution.
Newton's method for multiple roots.
Introduction of the concepts: attractor, repeller, attraction pool and their relationship with solving nonlinear equations.

5. Function approximation.
Power series (Taylor series).
Polynomial interpolation.
Interpolation assumptions.
Continuous and discrete case.
Interpolation task.
An example of "reverse engineering".
Uniqueness of interpolation.
Matrix, Lagrange and Newton methods. Divided differences.
Node selection problem. Equidistant and Chebyshev nodes.
Convergence of interpolation processes, Runge phenomenon.
Interpolation of polynomials - special cases.
Estimation of the error of the interpolation formula.

6. Numerical integration.
Interpolation in numerical integration.
Simple and complex Newton-Cotes quadratics.
Trapezoidal and Simpson's formulas.
Derivation of formulas.
Graphical interpretation of methods.
Numerical integration errors (quadratures).
Estimating errors.
Derivation of formulas determining the number of integration intervals depending on the given accuracy of the solution using complex methods.

The laboratory program covers the following topics:

Introduction to Matlab.

1. Floating-point arithmetic.
Rounding error representations of activities.
Machine accuracy.
Investigating the properties of floating-point arithmetic.
Floating point overflow and underflow.
Single and double precision real types.
Examples of unstable algorithms and ill-conditioned tasks.

2. Numerical differentiation: Taylor's formula.
The impact of method error and rounding of activities on the quality of the solution.

3. Numerical methods for solving initial value problems for first-order ordinary differential equations - selected one-step methods of the Runge-Kutta type. Investigating the relationship between the integration step and the solution convergence and method error. Stability and instability of solutions. The influence of the order of convergence of methods on the quality of the solution.

4. Numerical solution of nonlinear equations of functions of one variable.
Selected iterative methods: Tangent method (Newton-Raphson), bisection, secants and regula-falsi.
Exploring the relationship between the data and the obtained solution.
Exploring the relationship between termination conditions and solution accuracy.
Verification of the correctness of the solution.
Examples of method divergence.
Researching solutions for tasks that do not meet the required assumptions.

5. Function approximation.
Power series (Taylor series).
Polynomial interpolation.
Lagrange or Newton's method.
Study of the approximation of the selected function for equidistant and Chebyshev nodes - the problem of optimal selection of nodes.
Runge phenomenon.

6. Numerical integration.
Simple and complex trapezoidal and Simpson methods.
Testing the accuracy of solutions.

Teaching methods

Lectures:
• with multimedia presentation supplemented with examples given on the blackboard;
• conducted in an interactive manner with formulating questions for students;
• into account the students’ activity during the class when issuing the final grade;
• the lecture initiating the discussion;
• presented in connection with practice;
Laboratory classes:
• presented in connection with the current knowledge of students;
• into account different aspects of the issues presented;
• of a new topic preceded by a reminder of related content known to students in other subjects;
• supplemented with multimedia presentations;
• reports by the laboratory’s leader;
• in teams;
• experiments.

Bibliography

Basic:
• Fortuna, Macukow, Wąsowski, Metody numeryczne, WNT: PWN, 2017;
• Kincaid, Cheney, Analiza numeryczna, WNT 2006;

• Magnucka-Blandzi, Dondajewski, Gleska, Szyszka, Metody numeryczne w MatLabie. Wybrane zagadnienia, Wyd. Politechniki Poznańskiej 2013.

Additional:
• Burden, Faires, Numerical analysis, Prindle, Weber & Schmidt, Boston;
• Rosłoniec, Wybrane metody numeryczne z przykładami zastosowań w zadaniach inżynierskich, Oficyna Wydawnicza Politechniki Warszawskiej 2008.

Breakdown of average student's workload

	Hours	ECTS
Total workload	150	6,00
Classes requiring direct contact with the teacher	77	3,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	73	3,00