COURSE DESCRIPTION CARD - SYLLABUS

Course name
Mathematical analysis tools for ICT [S1MiKC1E>NAMdICT]

Course

Field of study
Microelectronics and Digital Communication

Year/Semester
1/2

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
English

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
15

Laboratory classes
15

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

2,00

Coordinators

dr hab. Grzegorz Oleksik
grzegorz.oleksik@put.poznan.pl

dr hab. inż. Damian Karwowski
damian.karwowski@put.poznan.pl

Lecturers

Prerequisites

The student should have mastered knowledge of mathematics in the areas of Mathematical Analysis, Algebra, and the basics of optimization methods for ICT, as well as basic programming skills in Python. They should also be able to conduct correct logical reasoning and understand the necessity of expanding their competencies.

Course objective

Familiarizing oneself with mathematical analysis tools and developing competencies in using its apparatus. Utilizing the capabilities offered by the Python language to analyze and solve problems in the field of mathematical analysis. Shaping the ability to describe and solve simple engineering problems using the discussed methods.

Course-related learning outcomes

Knowledge:

1. Possesses advanced knowledge of mathematics and physics, including mathematical analysis, algebra, probability calculus, and the basics of hypothesis testing, enabling the analysis and modeling of physical phenomena and technical processes in electronic and telecommunication systems. [K1_W01]
2. Has knowledge of software engineering tools, team programming techniques, and software development and testing methodologies. [K1_W05]
3. Knows the basic principles of occupational health and safety.[K1_W16]

Skills:
1. Is able to apply mathematical tools, including mathematical analysis, algebra, and probability calculus, to solve problems in the ICT field, particularly in signal analysis and processing. [K1_U03]
2. Can program in high-level languages, including using multithreading and multiprocessor systems. Is capable of writing and running programs that solve technical problems in ICT. Can consciously select programming languages for specific applications and use software engineering tools, including those supporting team-based programming. [K1_U06]
3. Is able to use software containerization tools and data visualization techniques. Can analyze, interpret, and present data in a way that enables its effective use in decision-making. [K1_U07]

Social competences:
1. Is aware of the limitations of their own knowledge and skills and understands the need for continuous learning. [K1_K01]
2. Understands the importance of knowledge in solving technical and organizational problems. Can independently acquire and update information, but in situations requiring specialized knowledge, knows how to seek expert opinions and collaborate with specialists to find the optimal solution. [K1_K08]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

1. Lecture
Written test. The test consists of several to a dozen questions (depending on the nature of the questions) and covers the content presented during the lectures. Passing threshold: 50% of points.
2. Laboratory
Final test. The test consists of several to a dozen tasks (depending on the nature of the questions) to be completed using Python programming. Passing threshold: 50% of points.
For passing the lecture and laboratory, the following percentage thresholds apply to individual grades: 2.0 (< 50%), 3.0 (50%-59%), 3.5 (60%-69%), 4.0 (70%-79%), 4.5 (80%-89%), 5.0 (90% and more). The students' answers (independently for each question/task) are scored by awarding a portion of the points allocated for the question/task that corresponds to the degree to which the answer given is correct.

Programme content

Real functions of one and multiple variables. Differential and integral calculus of functions of one and multiple variables. Ordinary differential equations. Laplace transform. Numerical, power, and Fourier series.

Course topics

Lecture:
1. Integral calculus of functions of multiple variables (double, triple, and line integrals with applications).
2. Differential equations (general, particular, and singular solutions, initial value problems).
3. Selected first-order ordinary differential equations (separable, homogeneous, Bernoulli, exact, and linear equations).
4. Selected second-order ordinary differential equations (linear equations with constant coefficients).
5. Laplace transform.
6. Numerical, functional, and power series (expanding functions into Fourier series).
Laboratory: Creating plots of functions of one and two variables using the Matplotlib and NumPy libraries. Using SymPy and SciPy packages to:
1. Compute double, triple, and line integrals.
2. Solve first-order ordinary differential equations (separable, homogeneous, linear, exact).
3. Solve second-order linear ordinary differential equations with constant coefficients.
4. Calculate the Laplace transform and use it to solve differential equations.

5. Study the convergence of numerical, power, and Fourier series, and expand functions into power and Fourier series.

Teaching methods

Lecture: Traditional lecture Problem-based lecture
Laboratory:
- Presentation
- Individual and group work
- Computational experiments

Bibliography

Basic:
1. M. Gewert, Z. Skoczylas, Analiza matematyczna 1 i 2. Definicje, twierdzenia wzory.
2. M. Gewert, Z. Skoczylas, Analiza matematyczna 1 i 2. Przykłady i zadania
3. M. Gewert, Z. Skoczylas, Równania różniczkowe zwyczajne. Teoria, przykłady i zadania.
4. W. Krysicki, L. Włodarski, Analiza matematyczna w zadaniach cz.1 i 2.
5. R. Johansson, Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib, Helion, 2021
6. V. Steinkamp, Python for Engineering and Scientific Computing: A Guide to Empowering Engineers and Scientists with Essential Python Tools and Practical Applications (Rheinwerk Computing) First Edition

Additional:
1. G.M. Fichtenholz, rachunek różniczkowy i całkowy, tom 1, 2 i 3.
2. J. Mikołajski, Z.Sołtysiak, Zbiór zadań z matematyki dla studentów studiów technicznych
3. Amit Saha, Matematyka w Pythonie. Algebra, statystyka, analiza matematyczna i inne dziedziny, Helion
4. P. Farell, A. Fuentes, The Statistics and Calculus with Python Workshop: A comprehensive introduction to mathematics in Python for artificial intelligence applications. Packt.

Breakdown of average student's workload

	Hours	ECTS
Total workload	60	2,00
Classes requiring direct contact with the teacher	30	1,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	30	1,00