COURSE DESCRIPTION CARD - SYLLABUS

Course name
Algorithm and data structures [S1MiKC1E>AiSD1]

Course

Field of study
Microelectronics and Digital Communication

Year/Semester
1/1

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
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

5,00

Coordinators

dr inż. Paweł Sroka
pawel.sroka@put.poznan.pl

Lecturers

dr inż. Michał Sybis
michal.sybis@put.poznan.pl

dr inż. Salim Janji
salim.janji@put.poznan.pl

Prerequisites

Student starting this course should have basic knowledge of the high school-level mathematics and physics. Moreover, a student should be acquianted with the use of a PC/notebook. Finally, student should understand the necessity to acquire a new knowledge and skills stemming from a chosen field of studies.

Course objective

The aim is to teach a student the ways of solving basic computational problems in engineering by building appropriate algorithms and using selected data structures, including introduction to selected numerical methods.

Course-related learning outcomes

Knowledge:
1. Has a solid knowledge of construction, description and implementation of computational algorithms.
2. Knows the selected data structures and basic algorithms used to perform specific tasks with these

structures.
3. Understands the idea of iterative algorithms and knows the selected numerical methods used in engineering.

Skills:
1. Is able to develop computational algorithms for solving basic computational problems in the area of mathematics and engineering, and is able to describe them in a graphical way.
2. Is able to implement these algorithms using selected programming language.
3. Is able to use the learned data structures for storing, organizing and searching of data.

Social competences:
1. Understands the need of further self-study, in particular in the field of algorithms and data structures.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

The knowledge acquired in the lectures is verified in form of a written exam. The exam comprises 6-10 open-ended questions that are graded (with points) differently. The exam is passed if at least 50% of the total score is obtained. The grading scale is as follows:
below 50% - 2,0
50% to 60% - 3,0
above 60% to 70% - 3,5
above 70% to 80% - 4,0
above 80% to 90% - 4,5
above 90% - 5,0.
The abilities acquired during the laboratories are verified with 8-13 exercises conducted in the classes, with selected algorithms and data structures implemented using a computer. Each exercise is evaluated with points based on the solutions uploaded by students. Additionaly, tests encompassing the discussed material may be conducted with the use of computers and software, complemented with written parts. The requirement to pass is to obtain at least 50% of the total score possible with exercises and tests. Additionally, the final grade can be also influenced with the evaluation of student's level of knowledge and skills required to conduct the laboratories and with the homeworks. The grading scale is as follows:
below 50% - 2,0
50% to 60% - 3,0
above 60% to 70% - 3,5
above 70% to 80% - 4,0
above 80% to 90% - 4,5
above 90% - 5,0.

Programme content

In this course theoretical and practical aspects of building and describing algorithms are taught, as well as their application in solving of basic mathematical, physical or telecommunication problems. The course topics cover also selected data structures and their use for storing and processing of information. Selected numerical methods used in electronics and telecommunications are also discussed.

Course topics

Lectures cover the following topics:
1. Introduction to algorithms - textual and graphical description, formulation of simple algorithms (4 h).
2. Basic rules of building iterative algorithms and use of recursion (4 h).
3. Computational complexit y of algorithms (1 h).
4. Simple data structures: arrays, lists and queues (4 h).
5. Tree and heap data structures (3 h).
6. Sorting and searching algorithms making use of selected data structures (4 h).
7. Basic numerical methods: solving nonlinear equations, solving systems of linear equations, calculation of derivatives and integrals, approximation and interpolation (10 h).
In the laboratories the following topics are covered:
1. Introduction to algorithms - formulation of simple algorithms and their description. Implementation of these algorithms using selected programming language (4 h).

2. Building and implementation of iterative algorithms (6 h).
3. Operation with arrays, lists and queues (6 h).
4. Implementation of tree and heap data structures using selected programming language (3 h).
5. Implementation of sorting and searching algorithms using selected data structures (3 h).
6. Implementation of selected numerical methods (4 h).
7. Tests and revision (4 h).

Teaching methods

Lecture: multimedia presentation supported with additional exercises/examples solved on a board or presented on a computer using dedicated software (implementation code). Tutorial-style classes are also considered.
Laboratories: practical exercises - students solve algorithmic problems formulated by the teacher using computers and installed software or in a written form, correct solutions are provided and explained by the teacher.

Bibliography

Basic:
Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Wprowadzenie do algorytmów", Wydawnictwo Naukowe PWN, 2024
Feliks Kurp, "Algorytmy : struktury danych i złożoność obliczeniowa", Helion, 2022

Additional:
Piotr Wróblewski, "Algorytmy w Pythonie : techniki programowania dla praktyków", Helion, 2022
Qingkai Kong, Timmy Siauw, Alexandre Bayen, "Python Programming and Numerical Methods - A Guide for Engineers and Scientists" (online)

Breakdown of average student's workload

	Hours	ECTS
Total workload	135	5,00
Classes requiring direct contact with the teacher	60	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	75	3,00