

Course content

The course introduces students to the fundamental concepts of data structures, covering the essential logical and physical structures of data and their implementation methods, as well as fundamental algorithms for each structure. Students will develop analytical skills to identify suitable data structures for solving problems, implement algorithms in code, and perform complexity analysis and optimization. This theoretical and practical course is critical in building a strong foundation for programming in real-world engineering contexts. The practical sessions are designed to enhance students' programming abilities through hands-on problem-solving and debugging.

Key topics include:

1. Introduction to Data Structures: Basic concepts, logical and physical data structures.
2. Fundamental Algorithms: Implementation and use of algorithms associated with various data structures.
3. Problem Solving: Analyzing requirements and selecting appropriate data structures.
4. Complexity Analysis: Evaluating and optimizing algorithmic performance.

Course objectives

Knowledge

1. Master foundational knowledge, principles, and methodologies of data structures.
2. Understand complexity analysis of algorithms and apply this knowledge to optimize performance.

Skills

1. Represent and solve problems using pseudocode and the C programming language.
2. Analyze and propose solutions for programming requirements using data structures and algorithms.
3. Compare, optimize, and debug algorithms and programs, producing accurate and efficient solutions.

Competencies

1. Identify critical algorithms and key lines of code for problem-solving based on data structures.
2. Correctly articulate and document programming problems and solutions.