
Course title: ALGORITHMS AND DATA STRUCTURES

ECTS credit allocation (and other scores): 4,5

Semester: autumn

Level of study: ISCED-6 - first-cycle programmes (EQF-6)

Branch of science: Natural sciences

Language: English

Number of hours per semester: 30 lectures + 30 classes = 60 hours

Course coordinator/ Department and e-mail: Erasmus coordinator Anna Szczepkowska/ WMil,
anna.szczepkowska@matman.uwm.edu.pl

Type of classes: classes and lectures

Substantive content

CLASSES:

Review of programming basics. Recurrence. Array operations, notation O , Ω , Θ . Basic sorting algorithms and comparison of their runtime. Stack, queue, list. Hashboard. Basic operations on binary trees, passing preorder, inorder, postorder. Graphic structures. Going in and out. Greedy programming.

LECTURES:

The concept of the algorithm based on a simple sorting algorithm. Algorithmic complexity. Notation O , Ω , Θ . The idea of dividing and conquer on the example of merge sort. Recurrence, solving recursive equations. Elementary data structures: lists, queues, stacks. Heap, priority queues, Heapsort. Quicksort sorting. Lower complexity limit for sorting by value comparison. Sorting in linear time. Hashboards, hash function. Collision resolution: chain method, open addressing. Trees, binary trees. Binary search trees, basic elementary operations for binary trees. Red-black trees. Graphic structures, searching in and out. Problems of the shortest paths and minimal stretches of trees. The idea of greedy algorithms: the problem of giving up the rest, backpacking, Huffman coding. Dynamic programming.

Learning purpose:

On completion of the study programme the graduate will gain:

Knowledge:

He knows the concept of algorithm and computational complexity, basic instructions of high-level language used in imperative programming, he knows methods of algorithm solving: recurrence, divide and conquer. He has an organized and theoretically based knowledge of algorithms and data structures (arrays, queues, lists, stacks, trees and graphs).

Skills:

A student is able to design and validate a program with respect to the complexity of algorithms and to save it in a high level language. The ability to analyse algorithmic problems, assess their complexity and the correctness of proposed procedures. He has the ability to apply basic data structures in solving algorithmic problems.

Social Competencies:

He understands the need for and knows the possibilities of continuous training.

Basic literature:

1) S. Dasgupta, C. Papadimitriou, U. Vazirani, , Algorytmy , wyd. PWN, 2018

Supplementary literature:

The allocated number of ECTS points consists of:

Contact hours with an academic teacher: 2,60 ECTS points,

Student's independent work: 1,90 ECTS points,