# Algorithms and Data Structures 1

Course description

## Basic Information

Field of study : Analytical Computer Science

Path : -

Organizational unit : Faculty of Mathematics and Computer Science

Education level : first-cycle

Form of studies : full-time studies

Study profile : general academic

Required : mandatory

Education cycle : 2022/23

Course code : UJ.WMIIANS.140.03340.22

Languages : Polish

Disciplines : Computer Science

ISCED Classification : 0613 Software and applications development and analysis

USOS Code : WMI.TCS.ASD1.OL

Course coordinator

Maciej Ślusarek

Course instructors

Maciej Ślusarek, Piotr Micek, Krzysztof Potępa

| | | |
|---|---|---|
| Period Semester 3 | Form of verification of learning outcomes<br>graded credit<br>Course format and hours<br>lecture: 30 laboratory classes: 30 | ECTS credits 5.0 |

## Learning outcomes for the course

| Code | Outcomes in the area of: | Directional learning outcomes | Verification methods |
|---|---|---|---|

| Code | Outcomes in the area of: | Directional learning outcomes | Verification methods |
|---|---|---|---|
| Knowledge – Student knows and understands: | | | |
| W1 | knows advanced data structures based on binary search trees: AVL trees, red-black trees, B-trees, heaps, splay trees and methods of their programming implementation | IAN_K1_W04, IAN_K1_W06, IAN_K1_W07, IAN_K1_W08, IAN_K1_W11 | graded credit, credit |
| W2 | has in-depth knowledge of algorithm construction techniques, in particular dynamic programming and the greedy method | IAN_K1_W06, IAN_K1_W07, IAN_K1_W08, IAN_K1_W09, IAN_K1_W10, IAN_K1_W11, IAN_K1_W12 | graded credit, credit |
| W3 | knows basic as well as selected advanced algorithms for many graph problems | IAN_K1_W06, IAN_K1_W07, IAN_K1_W09, IAN_K1_W10, IAN_K1_W11 | graded credit, credit |
| Skills – Student can: | | | |
| U1 | model problems presented in natural language using mathematical language and algorithmic concepts | IAN_K1_U01, IAN_K1_U03, IAN_K1_U05, IAN_K1_U06, IAN_K1_U07, IAN_K1_U08, IAN_K1_U10, IAN_K1_U11, IAN_K1_U17, IAN_K1_U21, IAN_K1_U22 | graded credit |
| U2 | design and implement algorithms using basic and selected advanced algorithmic techniques | IAN_K1_U06, IAN_K1_U07, IAN_K1_U08, IAN_K1_U10, IAN_K1_U11, IAN_K1_U17, IAN_K1_U21, IAN_K1_U22 | graded credit |
| U3 | test their program, look for errors and optimize it | IAN_K1_U05, IAN_K1_U11, IAN_K1_U18 | graded credit |
| Social competences – Student is ready to: | | | |
| K1 | formulate precise questions that serve to deepen or supplement their understanding of a given topic | IAN_K1_K01 | graded credit |

# ECTS credit balance

| Student's activity form | Average number of hours* allocated to completed activity types |
|---|---|
| lecture | 30 |
| laboratory classes | 30 |
| independently solving computer problems | 60 |
| preparation for classes | 30 |

| | | |
|---|---|---|
| Total student workload | Hours 150 | ECTS 5.0 |

* hour (lesson) means 45 minutes

## Course content

| No. | Course content | Learning outcomes for the course |
|---|---|---|
| 1. | 1. Dynamic programming: DAG of subtasks, solution reconstruction, memory size problem. Examples: traveling salesman problem, knapsack problem, longest common subsequence and Hirschberg's algorithm, optimal BST trees. 2. Greedy algorithms - selected examples: Huffman codes, scheduling with delay minimization, optimal buffering in cache memory. 3. Balanced trees: AVL trees, red-black trees, B-trees. 4. Other tree balancing mechanisms: probabilistic (heaps), amortized (splay trees). 5. Connectivity problems in graphs, strongly connected components, biconnected components. 6. Shortest paths in graphs, algorithms: Bellman/Ford, Dijkstra, Warshall/Floyd, Johnson. 7. Minimum spanning trees, algorithms: Jarnik/Prim, Boruvka/Sollin, Kruskal; disjoint-set data structure. 8. Network flows, algorithms: Ford/Fulkerson, Edmonds/Karp, push-relabel. 9. Bipartite graph matching, "turbo matching" algorithm, Hopcroft/Karp algorithm. | W1, W2, W3, U1, U2, U3, K1 |

## Extended information

### Teaching methods:

conventional lecture, laboratory classes

| Class type | Assessment forms | Course completion conditions |
|---|---|---|
| lecture | credit | Participation in lectures. |
| laboratory classes | graded credit | Laboratory completion based on assessment programs, homework assignments and tests. |

# Prerequisites and additional requirements

Programming Methods

# Literature

**Required**

1. T.H. Cormen, Ch.E. Leiserson, R.L. Rivest, C. Stein, Wprowadzenie do algorytmów, wydanie III, PWN, 2012

**Additional**

1. L.Banachowski, K.Diks, W.Rytter, Algorytmy i struktury danych, PWN, 2018