MODULE DESCRIPTION

Module code	full-time studies:	**Z-ZIP1-E-522**
	part-time studies:	**Z-ZIPN1-E-522**
Module name	**Algorithms and Data Structures**	
Module name in Polish	**Algorytmy i struktury danych**	
Valid from academic year	**2019/2020**	

MODULE PLACEMENT IN THE SYLLABUS

Field of study	**MANAGEMENT AND PRODUCTION ENGINEERING**
Level of education	**1st degree**
Studies profile	**General**
Form and method of conducting classes	**Full-time and Part-time**
Specialisation	**Computer Science for Management and Modelling**
Unit conducting the module	**Department of Computer Science Technologies**
Module co-ordinator	**Marcin Detka, PhD**
Approved by:	**Dariusz Bojczuk, PhD, DSc**

MODULE OVERVIEW

Type of subject / group of subjects	**Specialist subject**
Module status	**Compulsory**
Language of conducting classes	**English**
Module placement in the syllabus - semester	**Semesetr V**
Initial requirements	**Fundamentals of Computer Science Databases**
Examination (YES/NO)	**NO**
Number of ECTS credit points	**2**

Method of conducting classes		**Lecture**	**Classes**	**Laborato-ry**	**Project**	**Other**
Per semester	full-time studies:	**15**		**15**		
	part-time studies:	**9**		**9**		

TEACHING RESULTS AND THE METHODS OF ASSESSING TEACHING RESULTS

Category	Symbol	Learning outcomes	Assignations to the directional learning outcomes
Knowledge	W01	A student knows the methods of solving problems of an algorithmic nature, including computational. The student knows the basic data structures and their properties in the context of creating and building algorithms.	ZIP1 _W04 ZIP1 _W05
	W02	A student knows the rules for creating sequence algorithms and the student understands the principle of recursion and knows the advantages and risks of this method.	ZIP1 _W04 ZIP1 _W05
Skills	U01	A student has the capacity to select algorithms and data structures depending on the type and complexity of the problem. The student has the ability to select a method suitable for a computational problem.	ZIP1_U07 ZIP1_U14 ZIP1_U19
	U02	A student is able to formulate algorithms in a programming language and select appropriate data structures.	ZIP1_U07 ZIP1_U14 ZIP1_U19
Social competences	K01	He can supplement and improve the acquired knowledge and skills in the field of data structures and algorithms operating on these structures.	ZIP1_K01

TEACHING CONTENTS

Method of conducting classes	Teaching contents
Lecture	Basic principles of algorithm analysis: correctness, computational complexity of the algorithm: pessimistic, expected. Recursive algorithms. Notation of recursive algorithms in a programming language. Threats of recursive solutions. Recursion as hidden memory. Derecursivation. Numerical algorithms: numerical integration algorithms, numerical differentiation, solving nonlinear equations, solving systems of equations Basic data structures, stacks, lists, queues Sorting algorithms: sort by comparison (InsertionSort, QuickSort, MergeSort), simple priority queues: HeapSort binary heaps, positional sort, the complexity of the sort problem Search algorithms: linear search, binary search, hashing Text searching, brute-force algorithms, K-M-P algorithm, Boyer and Moore algorithm, Rabin and Karp algorithm Advanced programming techniques: dynamic programming, greedy algorithms

Laboratory	Notation of algorithms in different notations. Implementation in selected programming languages. Assessment of the correctness and computational complexity of algorithms. Implementation of recursive algorithms, incl. factorial, Fibonacci sequence, Euclid's algorithm, Hanoi towers others. Defining solutions to problems using recursion. Tracking recursive calls. Derecursivation of algorithms. Implementation of selected numerical algorithms. Assessment of the convergence of algorithms and the accuracy of calculations. Defining solutions to problems with the use of numerical algorithms Implementation of sorting algorithms. Generating test sets for algorithms. Measurement of algorithm execution time for various test sets. Implementation of selected searching algorithms. Comparison of the effectiveness of individual approaches to the search problem. Implementation and testing of text search algorithms. Assessment of the effectiveness of text search algorithms. Implementation of dynamic and greedy algorithms for selected problems, incl. the problem of cutting the rod, the knapsack problem and others. Assessment of the effectiveness of selected solutions.

METODS OF ASSESSING TEACHING RESULTS

Symbol	Methods of checking the learning outcomes *(select X)*					
	Oral exam	**Written exam**	**Test**	**Project**	**Statement**	**Other**
W01			X		X	
W02			X		X	
U01			X		X	
U02			X		X	
K01			X		X	

FORM AND CONDITIONS OF PASSING

Form of classes	Form of credit	Passing conditions
Lecture	Credit with grade	Obtaining 50% of points from a written final thesis, the scope of which applies to both lectures and laboratories.
Laboratory	Credit with grade	A student scores points for activity in laboratories, for the preparation of reports to selected laboratories (according to the instructor's indications) and for two practical tests at computers. The condition for passing is 50% of the points.

STUDENT WORKLOAD

<table>
<tr><th colspan="13">Balance of ECTS points</th></tr>
<tr><td rowspan="3">No.</td><td rowspan="3">Type of student's activity</td><td colspan="10">Student's workload</td><td rowspan="3">Unit</td></tr>
<tr><td colspan="5">full-time</td><td colspan="5">part-time</td></tr>
<tr><td>Lc</td><td>C</td><td>Lb</td><td>P</td><td>O</td><td>Lc</td><td>C</td><td>Lb</td><td>P</td><td>O</td></tr>
<tr><td>1.</td><td>Participation in the activities</td><td>15</td><td></td><td>15</td><td></td><td></td><td>9</td><td></td><td>9</td><td></td><td></td><td>h</td></tr>
<tr><td>2.</td><td>Other (consultation, exam)</td><td>2</td><td></td><td>2</td><td></td><td></td><td>2</td><td></td><td>2</td><td></td><td></td><td>h</td></tr>
<tr><td>3.</td><td>Number of hours of a student's assisted work</td><td colspan="5">34</td><td colspan="5">22</td><td>h</td></tr>
<tr><td>4.</td><td>Number of ECTS credit points which are allocated for assisted work</td><td colspan="5">1,4</td><td colspan="5">0,9</td><td>ECTS</td></tr>
<tr><td>5.</td><td>Number of hours of a student's unassisted work</td><td colspan="5">16</td><td colspan="5">28</td><td>h</td></tr>
<tr><td>6.</td><td>Number of ECTS credit points which a student receives for unassisted work</td><td colspan="5">0,6</td><td colspan="5">1,1</td><td>ECTS</td></tr>
<tr><td>7.</td><td>Work input connected with practical classes</td><td colspan="5">25</td><td colspan="5">25</td><td>h</td></tr>
<tr><td>8.</td><td>Number of ECTS credit points which a student receives for practical classes</td><td colspan="5">1,0</td><td colspan="5">1,0</td><td>ECTS</td></tr>
<tr><td>9.</td><td>Total number of hours of a student's work</td><td colspan="5">50</td><td colspan="5">50</td><td>h</td></tr>
<tr><td>10.</td><td>Punkty ECTS za moduł
1 ECTS=25 hours</td><td colspan="10">2</td><td>ECTS</td></tr>
</table>

LITERATURE

1. Cormen T. H., Leiserson C. E. , Rivest R. L., Stein C. (2022), Introduction to Algorithms, 4th edition, The MIT Press (https://dl.ebooksworld.ir/books/Introduction.to.Algorithms.4th.Leiserson.Stein.Rivest.Cormen.MIT.Press.9780262046305.EBooksWorld.ir.pdf)
2. Cormen T. H. (2013), *Algorithms Unlocked*, The MIT Press.
3. Wengrow J. (2020), *A Common-Sense Guide to Data Structures and Algorithms: Level Up Your Core Programming Skills*, 2nd edition, O'Reilly Media.