GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Algorithms and Data Structures, PG_00047652		
Field of study	Informatics		
Date of commencement of studies	October 2020	Academic year of realisation of subject	2020/2021
Education level	first-cycle studies	Subject group	Obligatory subject group in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	1	Language of instruction	Polish
Semester of study	2	ECTS credits	5.0
Learning profile	general academic profile	Assessment form	exam
Conducting unit	Department of Algorithms and Systems Modelling -> Faculty of Electronics, Telecommunications and Informatics		
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Krzysztof Manuszewski	
	Teachers	dr inż. Krzysztof Manuszewski dr Marcin Jurkiewicz mgr inż. Tomasz Goluch mgr inż. Andrzej Jastrzębski mgr inż. Kacper Wereszko	

Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	30.0	0.0	15.0	15.0	0.0	60
	E-learning hours included: 0.0						
	Adresy na platformie eNauczanie:						

Learning activity and number of study hours	Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM
	Number of study hours	60	5.0	60.0	125

Subject objectives	Major goal is introduction to algorithms and data structures. During the course there are presented some basic data structures and basic algorithms from various domains. Presented are tree data structures, hash tables, balanced trees, B-trees and joinable heaps. During the course there are presented basic ideas about construction of algorithms.

Learning outcomes	Course outcome	Subject outcome	Method of verification
	[K6_U43] can analyse date and formulate, apply and assess appropriate formal models and algorithms for solving problems in the field of information systems and applications	knowledge about basic data structures, ability to understanding and implementation algorithms of various complexity,	[SU1] Assessment of task fulfilment
	[K6_U09] can carry out a critical analysis of the functioning of existing technical solutions and assess these solutions, as well as apply experience related to the maintenance of technical systems, devices and facilities typical for the field of studies, gained in the professional engineering environment	skills in areas of problem analysis and model creation,	[SU1] Assessment of task fulfilment
	[K6_U04] can apply knowledge of programming methods and techniques as well as select and apply appropriate programming methods and tools in computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study	knowledge about basic data structures and dedicated algorithms, ability to understanding and implementation algorithms of various complexity,	[SU1] Assessment of task fulfilment
	[K6_U01] can apply mathematical knowledge to formulate and solve complex and non-typical problems related to the field of study and perform tasks, in an innovative way, in not entirely predictable conditions, by:n- appropriate selection of sources and information obtained from them, assessment, critical analysis and synthesis of this information,n- selection and application of appropriate methods and toolsn	Student knows the idea of precise and approximated algorithm. Student is able to adapt algorithm to the problem constraints	[SU1] Assessment of task fulfilment
	[K6_W41] Knows and understands, to an advanced extent, the operation and evaluation criteria of data processing, storage and transfer methods, including computational algorithms, artificial intelligence and data mining	student understands how to evaluate algorithm and has understanding the complexity idea. Student knows basic data structures and algorithms. Student knows the basic methods for algorithms construction	[SW1] Assessment of factual knowledge

Subject contents	Schema of problem solution: analysis of situation and analysis of goal, impact of model on solution Algorithmic problems, algorithms notation, analysis, correctness, stop, Estimation of function growth, *O* notation, time vs. complexity Examples if recursion/iteration, recursive and iterative algorithms Brute-force method, heuristic method, Dynamic programming Examples of recursion for algorithms based on strategy divide and conquer Basic data structures (list, queue, stack) and methods of their realization Simple sorting algorithms: insertion, selection, change. Binary search Sorting algorithms based on strategy divide and conquer Heap sort, Bucket sort and positional sort, , Search for *k*-th minimal element Adressing and hashing functions, Trees and algorithms for trees: BFS, DFS, Elementary graph algorithms Binary search trees, priority queues, "Red-black" trees, B-Trees, Graph representation, the shortest path in graphs: Dijkstra, Spanning tree in graphs: Prim and Kruskal algorithms
Prerequisites and co-requisites	Knowledge about fundamentals of programming

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
	40.0%	34.0%
	40.0%	33.0%
	40.0%	33.0%

Recommended reading

Basic literature	T. Cormen, Introduction to algorithms, MIT 1994
Supplementary literature	http://www.algorytm.org/
eResources addresses	

Example issues/ example questions/ tasks being completed	LAB: implementation of iterative and recursive approaches, implementation of simple sorting methods, hash tables solution for knapsack problem PROJ: Implementation ONP based calculator for string operations. implementation MInimax for simple games
Work placement	Not applicable