GDAŃSK UNIVERSITY
OF TECHNOLOGY

Subject card

Subject name and code	Discrete Optimisation Algorithms, PG_00048241		
Field of study	Informatics		
Date of commencement of studies	February 2024	Academic year of realisation of subject	2023/2024
Education level	second-cycle studies	Subject group	Optional subject group Subject group related to scientific research in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	1	Language of instruction	English
Semester of study	1	ECTS credits	2.0
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 hab. inż. Michał Małafiejski
	Teachers	mgr inż. Krzysztof Pastuszak dr hab. inż. Michał Małafiejski

Lesson types and methods of instruction	Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
	Number of study hours	15.0	0.0	15.0	0.0	0.0	30
	E-learning hours included: 0.0						

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	30	4.0	16.0	50

Subject objectives	Introduction to the methods of the construction of approximation algorithms to the computationally hard problems. The ability to prove guaranteed approximation ratios of algorithms for selected optimization problems. The construction of approximation schemes using pseudo-polynomial time algorithms based on dynamic programming. Overview of the significant problems of discrete optimization.

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K7_K02] is ready to provide critical evaluation of received content and to acknowledge the importance of knowledge in solving cognitive and practical problems	The student conducts a methodological analysis of the possibilities of solving discrete optimization problems.	[SK5] Assessment of ability to solve problems that arise in practice
[K7_W04] Knows and understands, to an advanced extent, the principles, methods and techniques of programming and the principles of computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study, and organisation of systems using computers or such devices	Student is able to find or construct by himself an efficient approximation algorithm for the proposed model.	[SW3] Assessment of knowledge contained in written work and projects
[K7_U41] can select methods of modelling and analysis of information systems and applications using selected elements of theoretical computer science and modern programming tools	Student identifies methods of designing algorithms. Student applies the method of construction fulfilment of the approximate algorithms.	[SU1] Assessment of task fulfilment [SU5] Assessment of ability to present the results of task
[K7_U01] can apply mathematical knowledge to formulate and solve complex and non-typical problems related to the field of study by:n- appropriate selection of source information and its critical analysis, synthesis, creative interpretation and presentation,n- application of appropriate methods and toolsn	Student analyzes real-world problems using discrete mathematical models.	[SU1] Assessment of task fulfilment [SU2] Assessment of ability to analyse information
[K7_W01] Knows and understands, to an increased extent, mathematics to the extent necessary to formulate and solve complex issues related to the field of study.	Student examines and analyzes the quality of the approximate algorithms.	[SW2] Assessment of knowledge contained in presentation

Subject contents

1. Rules of grading 2. Methods of designing of algorithms 3. K-approximation algorithms 4. Approximation schemes 5. Approximation scheme for knapsack problem 6. PSPACE-complete problems 7. Approximation algorithms for k-center problem 8. Approximation algorithms for covering problems 9. Layering algorithm for Vertex Cover 10. Greedy algorithms for shortest superstring problem 11. 4- and 3-approximation algorithms for shortest superstring problem 12. Approximation algorithms for Steiner Tree problem 13. Approximation algorithms for TSP 14. Approximation algorithms for scheduling problems 15. Construction of algorithms based on linear programming

Prerequisites and co-requisites

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
Exam	50.0%	100.0%

Recommended reading

Basic literature	Vazirani, Vijay V. "Approximation Algorithms" Wilson Robin J. "Introduction to Graph Theory" Christos H. Papadimitriou "Computational Complexity"
Supplementary literature	No requirements
eResources addresses	Adresy na platformie eNauczanie:

Example issues/ example questions/ tasks being completed

-

Work placement

Not applicable