COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Algorithm Design and Analysis	

Lecturer(s)	Department where the course unit is delivered
Coordinator: lect. dr. Valdas Dičiūnas **Other lecturers:** -	Department of Computer Science Faculty of Mathematics and Informatics Vilnius University

Cycle	Type of the course unit
1st (BA)	Compulsory

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	Spring semester	English

Prerequisites
Prerequisites: Calculus I, Discrete Mathematics, Informatics fundamentals.

Number of credits allocated	Student's workload	Contact hours	Individual work
5	140	68	72

Purpose of the course unit: programme competences to be developed

Purpose of the course unit:
to develop student's ability to design efficient algorithms for real world discrete problems as well as to estimate the complexity of algorithms and problems.

Generic competences:

- Ability to analyse and organise the information (GK1).
- Ability to apply the knowledge in practice (GK2).

Specific competences:

- Analysis and applications of continuous and discrete mathematical structures (SK4).
- Development of algorithms and their complexity evaluation (SK5).
- Programming (SK6).
- Mathematical and computer modeling (SK10).

Learning outcomes of the course unit: students will be able to	Teaching and learning methods	Assessment methods
analyze algorithms and estimate their complexity	Interactive lecture Problem-oriented teaching Individual reading Problem solving Project programming	6 homeworks (written) Project (program code, experiments and typed report) Exam (written)
analyze complexity of real world problems		
design efficient algorithms in practise		
distinguish between tractable and intractable problems		

Course content: breakdown of the topics	Contact hours							Individual work: time and assignments	
	Lectures	Tutorials	Seminars	Practice	Laboratory work	Practical training	**Contact hours**	**Individual work**	**Assignments**
1. Introduction to algorithm analysis. Algorithms and their properties. Measuring complexity of algorithms and problems. Counting techniques usefull in algorithm analysis. Growth of functions. Upper and lower complexity bounds for sorting.	6				10		**16**	**12**	Individual reading Problem solving Project: algorithm implementation, analysis, testing and written presentation
2. Combinatorial objects, their properties and their presentation by different data structures. Integers, sets, sequences, trees and graphs.	4				4		**8**	**6**	
3. Basic techniques for the design and analysis of efficient algorithms: divide-and-conquer, dynamic programming, backtracking, branch-and-bound, greedy and heuristic algorithms. Analysis of Matrix Multiplication, Knapsack and Travelling Salesman problems.	8				9		**17**	**12**	
4. Main graph algorithms and their analysis. Depth-first search and Breadth-first search. Minimum Spanning Tree problem. Algorithms of Prim and Kruskal. Shortest Path problem. Floyd-Warshall algorithm. Euler and Hamilton graphs.	6				6		**12**	**8**	
5. Complexity clases P and NP. Reduction techniques and NP-complete problems. Problems CIRCUIT-SAT, SAT, CLIQUE, VERTEX COVER, HAMILTON and TSP. Approximation algorithms. Approximation schemes FPTAS and PTAS.	8				2		**10**	**10**	
Programming project					1		**1**	**14**	
Exam (written)		2					**4**	**10**	
Total	**32**	**2**			**32**		**68**	**72**	

Assessment strategy	Weight %	Deadline	Assessment criteria
6 homeworks	30	During the semester, every 2-3 weeks	Each homework consists of 2-3 individual problems and is assessed by 0.5 point in total. **To be allowed to pass the exam a student must have at least 1.2 homework points (i.e., 40% of 3 points).**
Programming project: to implement a concrete algorithm, to test it on the problems of different size, to estimate algorithm complexity theoretically and practically, and to prepare a typed presentation with program demonstration.	30	May	Programming project is assessed by 3 points as follows: 0-1 point for algorithm implementation (source code); 0-0.5 point for experiments; 0-0.5 point for algorithm analysis; 0-1 point for typed presentation
Exam (written)	40	June	Exam consists of 5-6 theoretical questions and problems. Exam work is assessed by 4 points as follows: 4 – excellent knowledge and abilities; 3 – strong knowledge and abilities; 2 – mediocre knowledge and abilities; 1 – minimal knowledge and abilities; < 1 – minimal requirements are not satisfied. **To be allowed to pass the exam a student must have at least 2 points for his/her homeworks and project including no less than 40% of homework points.**

Author	Publishing year	Title	Number or volume	Publisher or URL
Required reading				
V. Dičiūnas	2012	Algorithm Analysis		**http://www.mif.vu.lt/~valdas/ALGORITMAI/Erasmus/Tutorial/manual.pdf**
Recommended reading				
T.H.Cormen, C.E. Leiserson, R.L. Rivest and C. Stein	2009	Introduction to Algorithms, 3rd edn.		MIT Press
E.M. Reingold, J. Nievergelt and N. Deo	1977	Combinatorial Algorithms: Theory and Practice		Prentice-Hall