



MODULIO APRAŠAS

Modulio pavadinimas	Kodas
Algoritmų kūrimas ir analizė	

Dėstytojas	Padalinys
Koordinuojantis: Valdas Dičiūnas Kitas (-i):	Informatikos katedra Matematikos ir informatikos fakultetas Vilniaus universitetas

Studijų pakopa	Dalyko tipas
Pirmoji	Individualiųjų studijų dalykas

Igyvendinimo forma	Vykdyto laikotarpis	Vykdyto kalbos
Auditorinė arba nuotolinio mokymo arba mišri	3 semestras	Lietuvių

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai: Diskrečios matematikos ir matematinės analizės (arba aukštosios matematikos) pagrindai, gebėjimas programuoti

Modulio apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5 ECTS	125	48	77

Modulio tikslas: studijų programos ugdomos kompetencijos		
Modulio tikslas – ugdyti studento analitinius įgūdžius, kurie leistų konstruoti efektyvius algoritmus realiems diskretiems uždaviniams spręsti ir savarankiškai analizuoti algoritmų bei uždavinių sudėtingumą.		
Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai
Gebės analizuoti algoritmus ir įvertinti jų sudėtingumą.	Įtraukiamoji paskaita Probleminis dėstymas Dalykinės literatūros studijavimas Uždavinių sprendimas Projektinė veikla	6 namų darbai (raštu) Projektinis darbas (algoritmo realizacija, eksperimentai, sudėtingumo analizė, aprašymas ir prezentacija) Egzaminas (raštu)
Gebės įvertinti realių praktinių uždavinių sudėtingumą.		
Gebės savarankiškai konstruoti efektyvius algoritmus realiems praktiniams uždaviniams.		
Gebės atskirti praktiškai nesunkiai išsprendžiamus uždavinius nuo neišsprendžiamų arba labai sudėtingų.		

Temos	Kontaktinio darbo valandos						Savarankiškų studijų laikas ir užduotys		
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai (LD)	Konsultavimas LD metu	Visas kontaktinis darbas	Savarankiškas darbas	Užduotys
1. Algoritmai ir jų sudėtingumas. Algoritmų ir uždavinių sudėtingumo matavimas. Funkcijų augimo greičiai. Viršutinis ir apatinis uždavinio sudėtingumo įverčiai.	4			8			12	12	
2. Kombinatoriniai objektai, jų savybės ir jų vaizdavimas įvairiomis duomenų struktūromis. Sveikieji skaičiai, medžiai ir grafai.	2			4			6	6	
3. Pagrindiniai algoritmų konstravimo metodai: skaldyk ir valdyk, dinaminis programavimas, paieška su grįžimu, šakų ir rėžių metodas. Kuprinės užpildymo bei Keliaujančio Pirklio uždavinių analizė.	4			8			12	12	
4. Pagrindinės grafų teorijos sąvokos. Paieška gilyn ir paieška platyn. Minimalaus aprėpties medžio uždavinys. Primo ir Kraskalo algoritmai. Trumpiausių kelių uždavinys. Floyd-Warshall algoritmas. Oilerio ir Hamiltono grafai.	4			8			12	12	
5. Sudėtingumo klasės P ir NP. Polinominė redukcija ir NP-pilni uždaviniai.	2			4			6	6	
6. Projektinis darbas								16	Algoritmo programinė realizacija, eksperimentai ir pristatymas.
7. Egzaminas ir pasiruošimas								13	
Iš viso	16			32			48	77	

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
6 namų darbai	40	Semestro metu	Kiekvieną namų darbą sudaro 2-4 individualūs uždaviniai. Bendra maksimali balų suma už visus namų darbus yra 4 balai. Studentas įgyja teisę laikyti egzaminą tik tuo atveju, jeigu jis išsprendžia ne mažiau kaip 40% namų darbų (t.y. gauna ne mažiau kaip 1,6 balo iš 4 galimų).
Projektinis darbas: realizuoti programiškai konkretų algoritmą, atlikti eksperimentus, teoriškai ir praktiškai įvertinti sudėtingumą bei paruošti darbo aprašą ir programos demonstraciją.	30	Lapkritis-gruodis	Projektinis darbas vertinamas 3 balais: 0-1 balo už realizaciją (programos kodą); 0-0.5 balo už eksperimentus; 0-0.5 balo už algoritmo analizę; 0-1 balo už aprašą ir prezentaciją.
Egzaminas (raštu)	30	Semestro gale	Egzamino testą sudaro 5-6 teoriniai klausimai ir užduotys. Egzaminas vertinamas 0-3 balais. Studentas įgyja teisę laikyti egzaminą tik tuo atveju, jeigu jis už projektinį darbą ir namų darbus gauna ne mažiau kaip 3 balus, iš kurių ne mažiau kaip 1,6 balo už namų darbus.

Reikalavimai dalyko vertinimui eksterno būdu	
Įvertinimas galimas eksterno būdu:	Taip, jei tenkina aukščiau išvardintus reikalavimus, suteikiančius teisę laikyti egzaminą.

Autorius	Leidimo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
Privalomoji literatūra				
V. Dičiūnas	2005	Algoritmų analizės pagrindai		http://www.mif.vu.lt/katedros/cs/Asmen/algoritmų_analize.pdf
Papildoma literatūra				
T.H.Cormen, C.E. Leiserson, R.L. Rivest and C. Stein	2009	Introduction to Algorithms, 3rd edn.		MIT Press
Sandeep Sen and Amit Kumar	2019	Design and Analysis of Algorithms: A Contemporary Perspective		Cambridge University Press
E.M. Reingold, J. Nievergelt and N. Deo	1977	Combinatorial Algorithms: Theory and Practice		Prentice-Hall
R. Čiegis	2007	Duomenų struktūros, algoritmai ir jų analizė		Vilnius: Technika



COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Algorithm Design and Analysis	

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

Cycle	Type of the course unit
1 st (BA)	Individualised Studies Subject

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face or remote	3 semester	Lithuanian

Prerequisites
Prerequisites: Knowledge of foundations of Discrete Mathematic and Calculus (or course in Higher Mathematics), ability to write program codes.

Number of credits allocated	Student's workload	Contact hours	Individual work
5 ECTS	125	48	77

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.		
Learning outcomes of the course unit: students will be able to	Teaching and learning methods	Assessment methods
analyze algorithms and estimate their complexity	Lecture Case study Problem-oriented teaching Individual reading Problem solving	Homeworks (written) Programming project (program code, experiments, typed report and presentation) 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 (LW)	Tutorial during LW	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.	4			8			12	12	
2. Combinatorial objects, their properties and their presentation by different data structures. Integers, sets, sequences, trees and graphs.	2			4			6	6	Project: algorithm implementation, analysis, testing and written presentation
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.	4			8			12	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.	4			8			12	12	
5. Complexity classes 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.	2			4			6	6	
Programming project								16	
Exam (written)								13	
Total	16			32			48	77	

Assessment strategy	Weig	Deadline	Assessment criteria
----------------------------	-------------	-----------------	----------------------------

	ht %		
6 homeworks	40	During the semester, every 2-3 weeks	Each homework consists of 2-4 individual problems and is assessed by 0.5-1 point in total (maximum 4 points for all homeworks). To be allowed to pass the exam a student must have at least 1.6 homework points (i.e., 40% of 4 points).
Programming project: to implement a concrete algorithm, to make experiments, to estimate algorithm complexity and to prepare a typed presentation and program demonstration.	30	November-December	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)	30	January	Exam consists of 5-6 theoretical questions and problems. Exam work is assessed by 0-3 points. To be allowed to pass the exam a student must have at least 3 points for his/her homeworks and project including no less than 40% of homework points.

Author	Publis hing year	Title	Number or volume	Publisher or URL
Required reading				
V. Dičiūnas	2005	Algoritmų analizės pagrindai		http://www.mif.vu.lt/katedros/cs/Asmen/algoritmu_analize.pdf
Recommended reading				
T.H.Cormen, C.E. Leiserson, R.L. Rivest and C. Stein	2009	Introduction to Algorithms, 3rd edn.		MIT Press
Sandeep Sen and Amit Kumar	2019	Design and Analysis of Algorithms: A Contemporary Perspective		Cambridge University Press
E.M. Reingold, J. Nievergelt and N. Deo	1977	Combinatorial Algorithms: Theory and Practice		Prentice-Hall
R. Čiegis	2007	Duomenų struktūros, algoritmai ir jų analizė		Vilnius: Technika