COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Algorithms and Data Structures	

Lecturer(s)	Department where the course unit is delivered
Coordinator: Saulius Ragaišis **Other lecturers:**	Department Software Engineering Faculty of Mathematics and Informatics Vilnius University

Cycle	Type of the course unit
First	Compulsory

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	2nd semester	Lithuanian

Prerequisites
Prerequisites: Procedural programming.

Number of credits allocated	Student's workload	Contact hours	Individual work
5	132	68	64

Purpose of the course unit: programme competences to be developed

Purpose of the course unit –to provide students with classic data structures and algorithms, to develop skills for the formalization of algorithms and their complexity assessment and comparison; solidify skills of disciplined programming.

Generic competences:

- Communication and collaboration *(GK1).*
- Life-long learning *(GK2).*

Specific competences:

- Knowledge and skills of underlying conceptual basis *(SK4).*
- Technological and methodological knowledge and skills, professional competence *(SK6).*

Learning outcomes of the course unit: students will be able to	Teaching and learning methods	Assessment methods
Use classic data structures and algorithms, knowing their essential features.	Lectures, discussions, individual reading, laboratory works, additional mini-assignments	Exam (written), laboratory works, additional mini-assignments
Select data structures and algorithms suitable the problem, evaluate their suitability.		
Assess complexity of an algorithm.		
Get first team programming skills.		
Apply key metrics and estimate personal efforts, will deepen the understanding of the need for disciplined programming.		

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**
Course structure and requirements	1						**1**	**1**	
Abstract data types (ADT).	1				6		**7**	**7**	
Personal software process (PSP): metrics and their application.	1				2		**3**	**3**	
Linear data structures: linked lists, stack, queue, deque.	2						**2**	**2**	
Trees. Binary search tree. Expected height of binary search tree	2						**2**	**2**	
Priority queue	1						**1**	**1**	
Heap	1						**1**	**1**	
Recursion. Arrangements, backtracking	1				8		**9**	**9**	
Complexity of an algorithm	1						**1**	**1**	
Search: sequential and binary search.	1						**1**	**1**	Individual reading, laboratory works, additional mini-assignments
Sorting: internal and external sorting, probabilistic sorting, algorithms (selection sort, insertion sort, bubble sort, Shell sort, quicksort, internal and external merge sort, radix sort, heap sort). Quicksort for selection of an element.	3					*8*	**3**	**3**	
Example of modeling program	1				8		**9**	**9**	
PSP estimation model	2				2		**4**	**4**	
Hash tables, Collision resolution	2						**2**	**2**	
Graphs	2				6		**8**	**8**	
AVL trees. Maximum height of AVL tree	2						**2**	**2**	
Black-red trees	2						**2**	**2**	
2-3 trees. B-tress	2						**2**	**2**	
NP-completeness	2						**2**	**2**	
Skip lists	2						**2**	**2**	
Exam (written)		2					**4**		2 hours for tutorial, 2 hours for exam
Total	**32**	**2**			**32**	*8*	**68**	**64**	

Assessment strategy	Weight %	Deadline	Assessment criteria
4 laboratory works	25-40	4th, 8th,11th and 14th week of semester	Types of laboratory works: 1. Abstract data types. 2. Arrangements. 3. Modeling. 4. Graphs. In the 3rd laboratory work, students should use at least 2 ADTs created by colleagues in the 1st laboratory work. Students should record time allocated when performing laboratory works, review code starting from 2nd laboratory work and us this data for efforts and time estimation for the 3rd and 4th laboratory works. Students should be able to explain and modify the programs. The penalty for exceeding the deadline is 20% for the each week. Laboratory works should be performed in defined order.

Additional mini-assignments	0-15	During laboratory work	Students, willing to collect additional points, may take optional mini-assignments (one mini-assignment per one laboratory work). There will be 10 mini-assignments in total, 0.15 points each.
Exam (written)	60	During exam session	A student can take part in the examination only if he/she has performed at least 3 laboratory works. The exam consists of theory questions and problem solving (of diverse difficulty).

Author	**Publishing year**	**Title**	**Number or volume**	**Publisher or URL**
Required reading				
Saulius Ragaišis	2007	Algorithms and data structures. Lectures material (in Lithuanian)		http://www.mif.vu.lt/~ragaisis/ADS2013/
Saulius Ragaišis	2007	Personal Software Process (in Lithuanian)		http://www.mif.vu.lt/~ragaisis/PSP2007/Asmeninis.programu.kurimo.procesas.pdf
Recommended reading				
Michael T. Goodrich, Roberto Tamassia	2002	Algorithm Design: Foundations, Analysis, and Internet Examples		John Wiley & Sons
Michael Main, Walter Savitch	2001	Data Structures and Other Objects Using C++		Addison Wesley
Mark Allen Weiss	1997	Data Structures and Algorithm Analysis in C	Second edition	Addison-Wesley
Michael Main, Walter Savitch	1995	Data Structures and Other Objects, A Second Course in Computer Science (Turbo Pascal Edition)		The Benjamin/Cummings Publishing Company
Paul Helman, Robert Veroff, Frank R. Carrano	1991	Intermediate Problem Solving and Data Structures, Walls and Mirrors		The Benjamin/Cummings Publishing Company
Daniel D. McCracken	1987	A second course in Computer Science With Pascal		John Wesley & Sons
Algimantas Juozapavičius	1997	Data structures and algorithms (in Lithuanian)		Vilnius University Publisher