MODULE DESCRIPTION

Module title	Module code
Parallel programming	

Lecturer(s)	Department where the module is delivered
Coordinator: dr. Algirdas Lančinskas **Other lecturers:**	Department of Software Engineering Faculty of Mathematics and Informatics Vilnius University

Cycle	Type of the module
First	Optional

Mode of delivery	Semester or period when the module is delivered	Language of instruction
Face-to-face	Autumn semester Third or Fourth year of study	Lithuanian

Prerequisites
Prerequisites: Object-oriented Programming II

Number of credits allocated	Student‘s workload	Contact hours	Self-study hours
5	130	68	62

Purpose of the module: programme competences to be developed

Purpose of the module – to give knowledge in parallel programming and principles of parallel computing systems, develop the ability to design and evaluate parallel algorithms of different types.

Generic competences:

- Life-long learning *(GK2)*.

Specific competences:

- Knowledge and skills of underlying conceptual basis *(SK4).*
- Software development knowledge and skills *(SK5).*
- Technological and methodological knowledge and skills, professional competence *(SK6).*

Learning outcomes of the module: students will be able to	Teaching and learning methods	Assessment methods
An ability to define parallel programming concepts, give examples.	Problem-oriented teaching Case analysis Solution of practical problems Individual reading	Laboratory assignments, examination in written form
An ability to design parallel algorithms suitable to different parallel computing systems.		
Will be familiar with different parallel programming standards and their applications.		
An ability to apply parallel programming methods to solve typical practical problems.		
An ability to undertake literature searches and analysis, apply obtained knowledge to solve practical problems.		

Content: breakdown of the topics	**Contact hours**							**Self-study work: time and assignments**	
	Lectures	Tutorials	Seminars	Practice	Laboratory work (LW)	*Tutorial during LW*	**Contact hours**	**Self-study hours**	**Assignments**
Concept and importance of parallel programming	2						**2**	2	Individual reading, problem solving laboratory assignments
Architectures of parallel computing systems	2				2	*1*	**4**	2	
Complexity of parallel algorithms, speed-up and efficiency coefficients	2				2	*1*	**4**	2	
Distributed memory parallel programming: MPI	8				8	*1*	**16**	10	
Shared memory parallel programming: OpenMP	6				6	*1*	**12**	10	
Shared memory parallel programming: POSIX Threads	6				6	*1*	**12**	10	
Scheduling tasks on parallel processors	2				4	*1*	**6**	4	
Review and analysis of applications of parallel programming to solve practical problems	2				2	*1*	**4**	4	
GRID technologies	2				2	*1*	**4**	2	
Preparation for the exam (exam is taken in written form)		2					**4**	16	2 hours for tutorial, 16 hours for preparation, 2 hours for the exam
Total	**32**	**2**			**32**	*8*	**68**	**62**	

Assessment strategy	**Weight %**	**Deadline**	**Assessment criteria**
Three laboratory assignments	50	Fourth, eighth and twelfth week of the semester	Practical problems are being solved in laboratory work. Students must demonstrate their ability to apply theoretical knowledge to analyze practical problems in order to propose an appropriate solution of the problem. In a case of delay, the score is reduced by one for every week of delay.
Exam (in written form)	50	During the exam session	Students which pass all laboratory assignments have the opportunity to take the exam. Students must answer theoretical questions during the exam. The maximum score is given for excellent theoretical knowledge, understanding and ability to give examples.

Author	**Publishing year**	**Title**	**Number or volume**	**Publisher or URL**
Required reading				
R. Čiegis	2001	Parallel algorithms (in Lithuanian)		Vilnius: Technika
Recommended reading				
R. Čiegis	2005	Parallel algorithms and network technologies (in Lithuanian)		Vilnius: Technika
B. Wilkinson, M. Allen	1999	Parallel Programming		Prentice-Hall
Gregory R. Andrews	2000	Foundations of Multithreaded, Parallel, and Distributed Programming		Addison Wesley