COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Parallel and Distributed Computing	**PMLS7124**

Lecturer(s)	Department where the course unit is delivered
Coordinator: Prof. Rimantas Vaicekauskas **Other lecturers:** -	Institute of Computer Science Faculty of Mathematics and Informatics Vilnius University

Cycle	Level of course unit	Type of the course unit
2nd (MA)	-	Compulsory

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	1st semester (5th semester part-time)	Lithuanian, English

Prerequisites and corequisites	
Prerequisites: -	**Corequisites (if any):** -

Number of credits allocated	Student's workload	Contact hours	Individual work
5	130	48	82

Purpose of the course unit: programme competences to be developed

Purpose of the course unit: to develop an understanding of the principles and paradigms of designing, analyzing, implementing and using of parallel high-performance software and hardware systems, algorithms and applications

Generic competences:

- Planning and organization *(GK2).*

Specific competences:

- Software requirements and design (SK4).
- Software construction (SK5).

Learning outcomes of the course unit: students will be able to	Teaching and learning methods	Assessment methods
- evaluate parallel computation methods and tools and choose the right system for developing a specific application.	Lectures, problem-oriented teaching, case studies, information retrieval, literary reading, individual work, laboratory work	Individual laboratory works and results presentation, written exam (open, semi-open and close-ended questions and tasks).
- formalize a subject area, to formally define the specific requirements and use them in the parallel computation		
- clearly present the chosen topic, summarize it, argue and defend his/her own opinion.	Information retrieval, literary readings, report preparation and presentation at the seminar, group discussion, demonstration	Presentation material, the oral presentation, answers to oral questions

Course content: breakdown of the topics	Contact hours							Individual work: time and assignments	
	Lectures	Tutorials	Seminars	Practice	Laboratory work	Practice	**Contact hours**	**Individual work**	**Assignments**
1. The fundamentals of parallel computations. Concurrency and synchronization, using shared memory variables and semaphores.	2						**2**	**4**	Self-study of literature.
2. Language level synchronization. Monitors. Java programming language.	2				2		**4**	**4**	Self-study of literature. Preparation for the 1st laboratory work.
3. Deadlock: detection, avoidance, prevention and recovery. Banker's algorithm. Mathematical modeling of deadlock using Petri nets.	2				2		**4**	**8**	Self-study of literature. Preparation for the delivery of 1st laboratory work.
4. Threads implementation: POSIX, OpenMP. Paradigms of shared memory computations: master-slave, work pool, pipelined computation.	2				2		**4**	**8**	
5. Parallel architectures, MIMD and SIMD classifications, network topologies, Amdahl's law, speed up and scalability. Performance analysis.	2				2		**4**	**8**	Self-study of literature. Preparation for the 2nd laboratory work. Prepare presentation.
6. Parallel programming using message passing tool MPI. Parallel Monte Carlo methods, Mandelbrot set computations.	2		2		2		**6**	**8**	
7. Partitioning and divide-and-conquer strategies. Gravitational N-Body problem. Barnes-Hut algorithm.	2		2		2		**6**	**8**	
8. Pipelined computations. Parallel sieve of Eratosthenes, solving system of linear equations.	2		2		2		**6**	**8**	
9. Synchronous computations. Jacobi iteration. Parallel algorithm for solving heat distribution problem.	2		2		2		**6**	**6**	
10. Load balancing and termination detection. Shortest path problem.	1		2				**3**	**6**	Prepare and deliver a presentation on a solving given (scientific) problem using parallel computations, which is not discussed in the study program. Self-study of literature.
11. Parallel algorithms and applications: sorting, numerical algorithms, image processing.	1		2				**3**	**6**	
12. Preparing for the exam and taking the final exam (written)								**8**	Self-study of literature, self-control tasks
Total	**20**		**12**		**16**		**48**	**82**	

Assessment strategy	Weight, %	Deadline	Assessment criteria
Oral presentation	30	During the semester	The following aspects of the presentation are assessed: - The presentation structure, size and style: the structure is clear and logical, contains all necessary components (introduction, explanation, conclusions), the presentation is of a reasonable duration; the material was delivered for a preview. - Completeness, recommendations and conclusions: The material presented in detail and in comparison to others methods/tools, recommendations and conclusions are grounded - 2 points; if the material is incomplete or the given conclusions are unreasonable - 1 point.
1st laboratory work		8th week of the semester	1st laboratory work: to create computer program that implements synchronization object to solve given coordination problem among threads, using *synchronized* statement (part A) and events (part B). The assessment of the laboratory work: A synchronization object is working correctly as described. The work is completed and defended on time (within 8 weeks) - 1 point. Lateness no more than 2 weeks leads to reducing the assessment in 25%, lateness no more than 4 weeks – 50%, later – 75%.
2nd laboratory work	20	16th week of the semester	2nd laboratory work: apply parallel programming tool (part A – OpenMP / POSIX threads, part B –MPI) to effectively solve specified computation problem. To evaluate mathematically and test practically the performance of the parallel solution. The assessment of the laboratory work: The parallel solution is working; it solves given problem and outputs the correct result. The presented performance analysis agrees with testing results. The results show increasing of the speed up with increasing number of processors used in the computation. The solution is completed and defended – 1 point.
Exam (written)	50	Exam session	The exam consists of 10-20 open, semi-open and close-ended questions and tasks each of them is assessed between 0.1 and 0.5 points. The exam is allowed only after presenting the report of the seminar and the collection of at least 2 points before the exam. The assessment of the exam: - 5 points: excellent knowledge and skills, the assessment level, collected at least 4.5 points; - 4 points: good knowledge and skills, the synthesis level, collected at least 3.5 points; - 3 points: average knowledge and skills, the analysis level, collected at least 2.5 points; - 2 points: knowledge and skills are less than average, the application level, collected at least 1.5 points. - 1 point: knowledge and skills are too low, collected less than 1.5 points.

Author	Year	Title	Number or volume	Publisher or URL
Required reading				
B. Wilkinson, M. Allen	2004	Parallel programming: techniques and applications using networked workstations and parallel computers	2nd ed.	Pearson
Ian Foster	1995	Designing and Building Parallel Programs	V1.3	http://www.mcs.anl.gov/~itf/dbpp/
Recommended reading				
Gregory R. Andrews	1999	Foundations of Multithreaded,	1st ed.	Addison - Wesley

		Parallel, and Distributed Programming		
By Raimondas Ciegis, David Henty, Bo Kagstrom and Julius Zilinskas	2008	Parallel Scientific Computing and Optimization: Advances and Applications (Springer Optimization and Its Applications)	1st. ed.	Springer