



MODULIO APRAŠAS

Modulio pavadinimas	Kodas
Lygiagretusis programavimas	

Dėstytojas	Padalinys
Koordinuojantis: doc. dr. Algirdas Lančinskas	Programų sistemų katedra, Informatikos institutas Matematikos ir informatikos fakultetas Vilniaus universitetas
Kitas (-i):	

Studijų pakopa	Dalyko tipas
Pirmoji	Pasirenkamasis

Igyvendinimo forma	Vykdymo laikotarpis	Vykdymo kalbos
Auditorinė	Rudens semestras (V ir VII)	Lietuvių

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai: Objektinis programavimas II

Modulio apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5	130	68	62

Modulio tikslas: studijų programos ugdomos kompetencijos		
Modulio tikslas – supažindinti studentus su lygiagrečiojo programavimo ir lygiagrečiųjų skaičiavimų sistemų veikimo principais ir svarba, ugdyti gebėjimą sudaryti ir vertinti įvairių tipų lygiagrečiuosius algoritmus.		
Bendrosios kompetencijos: - Nuolatinis mokymasis (BK2). - Gebės atlikti literatūros paiešką ir analizę, naudoti duomenų bazes ir kitus informacijos šaltinius (BK2.2). - Gebės savarankiškai įsisavinti naujas žinias, metodus ir įrankius bei taikyti juos praktikoje (BK2.3).		
Dalykinės kompetencijos: - Konceptualių pagrindų žinios ir gebėjimai (DK4). - Supras pagrindines programų sistemų inžinerijos koncepcijas bei sąvokas, įskaitant kelias priešakines sritis, suvoks galimas taikymo sritis ir žinos disciplinos aprėptį (DK4.1). - Gebės taikyti matematikos pagrindų, mokslo, inžinerijos, kompiuterių mokslo teorines žinias ir algoritminius principus programų sistemų kūrimo (DK4.2). - Gebės abstrakčiai mąstyti, naudoti formalius aprašymo metodus, įrodinėti jų teisingumą, formalizuoti ir specifiuoti realaus pasaulio problemas (DK4.3). - Programų sistemų kūrimo žinios ir gebėjimai (DK5). - Gebės išvelgti naujas programų sistemų taikymo galimybes, įvertinti taikomosios srities žinių poreikį, problemų kompleksumą bei jų sprendimų būdų įgyvendinamumą (DK5.1). - Gebės analizuoti problemą, identifikuoti poreikius ir apibrėžti reikalavimus tinkamam sprendimui (DK5.2). - Technologinės, metodinės žinios ir gebėjimai, profesinis kompetentingumas (DK6). - Gebės derinti teoriją ir praktiką programų sistemų taikymo įvairiose srityse uždavinių sprendimui, įvertinant technologinį, ekonominį, socialinį ir teisinį kontekstą (DK6.1). - Gebės panaudoti esamą kompiuterių techninę ir programinę įrangą, identifikuoti, perprasti ir taikyti perspektyvias technologijas (DK6.3). - Gebės planuoti, projektuoti ir atlikti eksperimentus bei kitus atitinkamus praktinius tyrimus, analizuoti ir interpretuoti duomenis (DK6.4).		
Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai

Gebės apibrėžti pagrindines lygiagrečiojo programavimo sąvokas, iliustruoti pavyzdžiais.	Probleminis dėstymas Atvejų analizė Praktinių užduočių sprendimas Savarankiškas literatūros analizavimas	Laboratoriniai darbai, egzaminavimas raštu
Gebės sudaryti lygiagrečiuosius algoritmus skirtingoms lygiagrečiųjų skaičiavimų sistemoms.		
Supras skirtingų lygiagrečiojo programavimo standartų skirtumus ir taikymo galimybes.		
Gebės taikyti lygiagrečiojo programavimo metodus tipiniams uždaviniams spręsti.		
Gebės atlikti literatūros paiešką ir analizę, tikyti įgytas žinias sprendžiant praktikoje pasitaikančius uždavinius.		

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
Lygiagrečiojo programavimo samprata ir svarba	2						2	2	Literatūros studijavimas, uždavinių sprendimas laboratorinių darbų metu ir namuose
Lygiagrečiųjų kompiuterių architektūra	2				2	1	4	2	
Lygiagrečiųjų algoritmų sudėtingumas, spartinimo ir efektyvumo koeficientai	2				2	1	4	2	
Paskirstytosios atminties lygiagretusis programavimas: MPI	8				8	1	16	10	
Bendrosios atminties lygiagretusis programavimas: OpenMP	6				6	1	12	10	
Bendrosios atminties lygiagretusis programavimas: POSIX Threads	6				6	1	12	10	
Užduočių sprendimo tvarkaraštis	2				4	1	6	4	
Taikomųjų uždavinių, kurių sprendimui naudojami lygiagretieji algoritmai, apžvalga ir analizė	2				2	1	4	4	
GRID technologijos	2				2	1	4	2	
Pasiruošimas egzaminui ir egzaminų laikymas		2					4	16	2 val. konsultacijoms, 16 val. pasiruošimui, 2 val. egzaminui
Iš viso	32	2			32	8	68	62	

Reikalavimai dalyko vertinimui eksterno būdu	
Įvertinimas galimas eksterno būdu:	Taip
Studentui turi atlikti dėstytojo pateiktus laboratorinius darbus ir laikyti egzaminą.	
Maksimalus balas skiriamas už puikų teorinių žinių įsisavinimą ir gebėjimą taikyti praktikoje.	

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
Trys laboratoriniai darbai	50	Ketvirta, aštunta ir dvylikta semestro savaitės	Laboratorinių darbų metu sprendžiamos praktinės užduotys. Studentams pateikiamos užduotys, kurias sprenddami turi parodyti, kaip geba taikyti teorines žinias praktiniams uždaviniams spręsti, jas analizuoti ir pasiūlyti tinkamą(-us) sprendimą(-us) bei tvarkingai ir raštingai apiforminti darbą. Už kiekvieną pavėluotą savaitę vertinimas mažinamas 1 balu.

Egzaminas raštu	50	Egzaminų sesijos metu	Egzaminą leidžiama laikyti atsiskaičius visus laboratorinius darbus. Egzamino metu studentai atsakinėja į teorinius klausimus. Maksimalus balas skiriamas už puikų teorinių žinių įsisavinimą, gebėjimą operuoti sąvokomis, pateikti pavyzdžių.
-----------------	----	-----------------------	---

Autorius	Leidimo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
Privalomoji literatūra				
R. Čiegis	2001	Lygiagretieji algoritmai		Vilnius: Technika
Papildoma literatūra				
R. Čiegis	2005	Lygiagretieji algoritmai ir tinklinės technologijos		Vilnius: Technika
B. Wilkinson, M. Allen	1999	Parallel Programming		Prentice-Hall
Gregory R. Andrews	2000	Foundations of Multithreaded, Parallel, and Distributed Programming		Addison Wesley



MODULE DESCRIPTION

Module title	Module code
Parallel programming	

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

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 (GC2). - An ability to undertake literature searches and analysis, and to use data bases and other sources of information (GC2.2). - An ability independently to acquire new knowledge, methodologies, and tools and to apply them in practice (GC2.3). Specific competences: - Knowledge and skills of underlying conceptual basis (SC4). - Knowledge and understanding of the key aspects and concepts of software engineering, including some at the forefront of the discipline, insight into possible application fields, and an awareness of the wider spectrum of the discipline (SC4.1). - An ability to apply mathematical foundations, knowledge of science and engineering, computer science theory, and algorithmic principles in software systems development (SC4.2). - An ability to reason at abstract level, to use formal notation, to prove the correctness, and to apply formalisation and specification for real-world problems (SC4.3). - Software development knowledge and skills (SC5). - An ability to become familiar with new software engineering applications, to appreciate the extent of domain knowledge, to evaluate the complexity of the problems and the feasibility of their solution (SC5.1). - An ability to analyse a problem, identify needs and define the computing requirements appropriate to its solution (SC5.2). - Technological and methodological knowledge and skills, professional competence (SC6). - An ability to combine theory and practice to complete software engineering tasks from different application areas while considering the existing technical, economic, and social context (SC6.1).

◦ An ability to use existing hardware, software and application systems, to identify, understand and apply promising technologies (SC6.3). ◦ An ability to plan, design and conduct experiments and other appropriate practical investigations (e.g. of system performance), as well as to analyse and interpret data (GC6.4).		
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
Concept and importance of parallel programming	2						2	2
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
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	Publis hing 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