



SD(M) pavadinimas	Course title
Lygiagretusis programavimas	Parallel Programming

SD(M) priklausomybė studijų pakopai
Course subjection to study level

Studijos: Studies:	B – Pirmosios pakopos First cycle
-----------------------	--------------------------------------

SD(M) priklausomybė studijų programai
Course subjection to programme

SD(M) priklausomybė studijų krypčių ir krypčių grupei

The list of study fields and groups of fields

SD(M) priklausomybė dalykų grupei * Course subjection to group	1 – studijų dalyko Course	Studijų krypčių grupės kodas Code of the group of study fields	Studijų krypties kodas Code of the study field
SD(M) priklausomybė programos daliai ** Course subjection to part of the programme	B – Studijų krypties dalykų dalis Part of Study area Subjects		
Struktūrinė SD priklausomybė *** Course structural subjection	K – katedros Department	B	B01

*) Grupė: *) 1 - studijų dalyko; 2 - praktikos; 3 - baigiamojo darbo ar projekto; 4 - baigiamojo egzamino; 5 - tiriamojo darbo; 6 - profesinio testavimo; 7 - kitas.

**) A - Bendrųjų universitetinių studijų; B - Studijų krypties; C - Specializacijos.

***) U - universiteto; F - fakulteto; K - katedros.

*) Group: *) 1 - Course; 2 - Practice; 3 - Final Work or Project; 4 - Final Examination; 5 - Research Work; 6 - Professional Testing; 7 - Other.

**) A - General; B - Field; C - Specialization.

***) U - University; F - Faculty; K - Department.

SD(M) kodas

Course number

Fakultetas Faculty	Katedra Department	Pakopa * Study cycle	Modulio Nr. Number
F	M	B	16603

SD(M) kreditai

Course volume in credits

Iš viso: Total:	Iš jų: KD, KS, KP, PR There out:
3	0

SD(M) Atsiskaitymo forma

Course assessment

I, E1, E2, E, BE, BD, TD, A	KD, KS, KP, PR
E	-

*) B - pirmoji pakopa; A - vientisosios studijos; M - antroji pakopa.

*) B - first cycle studies; A - integrated studies; M - second cycle studies.

SD(M) valandų paskirstymas pagal studijų formas ir būdus

Distribution of course hours by study forms and ways

Studijų forma Study form	Valandos Hours								Kontaktinių Contact
	Kodas Code	Studijų būdas * Study way	Paskai- toms Lectures	Lab. darbams Laboratory works	Praty- boms Practical works	Konsul- tacijoms Consultati on	Sav. darbui Independent work	Iš viso Total	
Nuolatinės studijos Full-time studies	NL	S	24	24	0	2	30	80	50

*) Studijų būdas: S - semestrais; M - moduliais; C - ciklais; T - nuotolinis; NI - neakivaizdinis intensyvusis.

*) Study process forms: S - semesters; M - modules; C - periods; T - distance; NI - part-time.

SD(M) ANOTACIJA

Lygiagrečiųjų kompiuterių, klasterių ir paskirstytų sistemų architektūra. Pagrindiniai naudojamų tinklų tipai ir jų charakteristikos. Pagrindiniai lygiagrečių algoritmų sudarymo principai. Bendrosios ir paskirstytosios atminties programavimo modeliai. Lygiagrečių programų struktūra, programų komponentai. Lygiagrečios gijos, procesai, darbų paskirstymas, komunikavimas, sinchronizavimas. Lygiagretaus programavimo kalbos, MPI, OpenMP, CUDA programavimo priemonės.

Studentai numatytu tvarkaraštyje metu privalo atlikti ne mažiau kaip 80 proc. laboratorinių darbų.

ANNOTATION OF COURSE

Parallel and distributed computing architectures. Main processor networks and their characteristics. Main principles and methods of design of parallel algorithms. Shared and distributed memory parallel programming models. Parallel threads and processes. Generation, communication, synchronization. Parallel programming languages, libraries, and tools. OpenMP. MPI. GPGPU computing. CUDA.

Students must attend at least 80% of the time scheduled laboratory works.

SD(M) TIKSLAS

Dalyko tikslas - supažindinti studentus su lygiagrečiais algoritmais, jų sudarymo ir jų realizavimo principais bei lygiagrečių programavimo technologijomis: daugelio gijų, daugelio procesorių, paskirstytų sistemų.

AIM OF COURSE

The course aims to provide knowledge of modern parallel computing technologies, main methods for the design of parallel algorithms and parallel programming tools.

Pagrindinė literatūra (ne daugiau kaip 5 šaltiniai):

Main references (not more than 5 references)

Eil. Nr. No.	Leidinio autoriai ir pavadinimas (elektroninių leidinių ir žiniatinklio adreso) Authors and title (site address in case of e-publication)
1.	R. Čiegis. Lygiagretieji algoritmai ir tinklinės technologijos. Vilnius, Technika, 2005.
2.	Bertil Schmidt, Jorge Gonzalez-Dominguez, Christian Hundt, Moritz Schlarb. Parallel Programming: Concepts and Practice. Morgan Kaufmann, 2017.
3.	J. J. Dongarra, I. Foster, G. C. Fox, W. Gropp, K. Kennedy, L. Torczon, A. White. Source Book of Parallel Computing. Morgan Kaufmann, 2003.
4.	D. Storti, M. Yurtoglu. CUDA for Engineers: An Introduction to High-Performance Parallel Computing. Addison-Wesley Professional, 1st Edition, 2015.
5.	B. Barney. Introduction to Parallel Computing. Lawrence Livermore National Laboratory. https://computing.llnl.gov/tutorials/parallel_comp/

*) Kortelės pildymo metu

*) At the form filling moment

Papildoma literatūra (ne daugiau kaip 10 šaltinių):

Additional references (not more than 10 references)

Eil. Nr. No.	Leidinio autoriai ir pavadinimas (elektroninių leidinių ir žiniatinklio adreso) Authors and title (site address in case of e-publication)
1.	M. Ben-Ari. Principles of Concurrent and Distributed Programming. Addison-Wesley Publishing Company, 2006.
2.	V. Kumar, A. Grama, A. Gupta, G. Karypis. Introduction to Parallel Computing: Design and Analysis of Algorithms. The Benjamin/Cummings Publishing Company, 2003.

*) Kortelės pildymo metu

*) At the form filling moment

Reikalingi IT resursai * (nurodyti 1-3 alternatyvas, pageidautina, kad bent 1 būtų nemokama)

Required IT Resources

Eil. Nr. No.	Programinės įrangos pavadinimas, gamintojas Name of the software, manufacturer	Licencijos tipas (pagal įsigijimo būdą) License type
1	C++	Mokama, akademinė Paid, academic
2	C	Mokama, akademinė Paid, academic

*) Pildoma, jei tokie resursai reikalingi. Stulpelyje Licencijos tipas pasirenkamas iš sąrašo:

Mokama, akademinė

Mokama, komercinė

Nemokama

*) Should be completed if such reassures are needed. License type - select from the list:

Paid, academic

Paid, commercial

Savarankiško darbo turinys

Content of individual work

Savarankiško darbo turinys

Content of individual work

Užduoties pavadinimas <i>Assignment title</i>	Sav. darbo apimtis vienai užduočiai <i>Amount of hours of independent work for a single task</i>						Užduočių skaičius <i>Number of tasks</i>					Iš viso valandų <i>Total hours</i>					Įvertinimo dalis % <i>Part of Evaluation %</i>							
	Rekomenduojamos val. <i>Recommended hours</i>	Skirta val. <i>Separated hours</i>					NL (T)	NL (S)	NL (Sav.)	I(S)	I(T)	NL(T)	NL(S)	NL (Sav.)	I(S)	I(T)	NL(T)	NL(S)	NL (Sav.)	I(S)	I(T)			
		NL(T)	NL(S)	NL (Sav.)	I(S)	I(T)																		
Kolokviumas Intermediate examination	8-27		8					1					8						20					
Namų darbas Home work	4-27		4					2					8						20					
Laboratorinis darbas Laboratory work	2-12		4					1					4						10					
Pasirengimas atsiskaitymui Preparation for evaluation	10-60		10					1					10											
Iš viso: <i>Total:</i>													30											

*) Papildomas laukas pildomas tik tada, kada taikomas SD(M) kortelėje nenurodytas studijų būdas: M - moduliai; C - ciklais; T - nuotolinis

*) Must be used in case study way does not fall into standard category: M - modules; C - periods; T - distance

Savarankiško darbo grafikas

Individual work schedule

Užduoties tipas <i>Task type</i>	Užduoties pateikimo(*) ir atsiskaitymo(+) savaitė <i>Week of Assignment setting (*) and assessment(+)</i>																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Nuolatinės studijos (S) <i>Full-time studies</i>																				
Kolokviumas <i>Intermediate examination</i>	*				1															
Namų darbas <i>Home work</i>	+		1					1												
Laboratorinis darbas <i>Laboratory work</i>	*	1									1					2				
	+									1										

Laboratorinių darbų sąrašas

List of the Course laboratory work

Temos pavadinimas <i>Topic title</i>	Valandų skaičius <i>Number of hours</i>				
	NL(T)	NL(S)	NL(Sav.)	I(S)	I(T)
1. Spartinančios (angl. "cache") atminties panaudojimo efektyvumo tyrimas matricos ir vektorių sandaugos algoritmų pagalba. Kompiliatorių optimizacijos tyrimas. BLAS bibliotekų panaudojimas. Darbas lygiagretaus klasterio Linux terpėje. <i>Investigation of cache memory performance using matrix-vector multiplication. Compiler optimization. Application of BLAS libraries. Job preparation and submission on Linux cluster.</i>		6			
2. Lygiagrečiųjų algoritmų realizavimas OpenMP pagalba. Tiesinės algebros uždavinių sprendimas. Skaičiavimo eksperimentų atlikimas. Spartinimo ir efektyvumo koeficientų vertinimas ir analizė. <i>Parallel solution of linear algebra problems. Implementation of parallel algorithms with OpenMP. Speed-up and efficiency analysis.</i>		6			
3. Lygiagrečiųjų algoritmų realizavimas MPI bibliotekos pagalba. Tiesinės algebros uždavinių sprendimas. Skaičiavimo eksperimentų atlikimas. Spartinimo ir efektyvumo koeficientų vertinimas ir analizė. <i>Parallel solution of linear algebra problems. Implementation of parallel algorithms with MPI. Speed-up and efficiency analysis.</i>		6			
4. Lygiagrečiųjų algoritmų realizavimas CUDA platformos pagalba. Efektyvus atminties panaudojimas. Optimizavimo technikų taikymas. CUDA bibliotekų panaudojimas taikomiesiems uždaviniams. <i>Implementation of parallel algorithms on CUDA platform. Efficient memory usage and optimization techniques. Application of CUDA libraries for practical problems.</i>		6			
Iš viso: <i>Total:</i>		24			

Paskaitų temų sąrašas
List of the Course lecture topics

Temos pavadinimas <i>Topic title</i>	Valandų skaičius <i>Number of hours</i>				
	NL(T)	NL(S)	NL(Sav.)	I(S)	I(T)
1. Lygiagrečiųjų kompiuterių ir paskirstytų sistemų architektūra. Pagrindinės klasifikacijos. Pagrindiniai tinklų tipai ir jų savybės. <i>Architectures of parallel and distributed computing systems. Main types and classifications, taxonomy. Main processor networks and their characteristics.</i>		4			
2. Bendrosios atminties lygiagretusis programavimas. Gijų valdymas ir sąveika. Gijų programavimo bibliotekos. <i>Shared memory parallel programming model. Threads programming: main principles, constructs and main libraries.</i>		4			
3. OpenMP programavimo standartas: pagrindiniai principai ir konstrukcijos. <i>Shared memory parallel programming with OpenMP: main principles and constructs.</i>		4			
4. Paskirstytosios atminties lygiagretusis programavimas. MPI programavimo biblioteka. Pagrindinės MPI funkcijos. <i>Distributed memory parallel programming model. Main principles, constructs and functions of MPI (Message passing Interface).</i>		4			
5. Grafinių procesorių panaudojimas bendros paskirties skaičiavimuose. <i>General-purpose computing on graphics processing units (GPGPU). GPU architectures.</i>		2			
6. CUDA platforma. CUDA programavimo modelis. <i>CUDA platform. CUDA programming model: main concepts and constructs. CUDA libraries.</i>		2			
7. Lygiagrečiųjų algoritmų analizė. Algoritmo spartinimo ir efektyvumo vertinimas. <i>Analysis of parallel algorithms. Speed-up and efficiency coefficients. Scalability of parallel algorithms.</i>		2			
8. Pagrindiniai lygiagrečių algoritmų sudarymo būdai. Duomenų lygiagretumas. Funkcinis lygiagretumas. <i>Main principles of design of parallel algorithms. Decomposition and mapping techniques.</i>		2			
Iš viso: <i>Total:</i>		24			

*) Papildomas laukas pildomas tik tada, kada taikomas SD(M) kortelėje nenurodytas studijų būdas: M - moduliai; C - ciklais; T - nuotolinis

*) Must be used in case study way does not fall into standard category: M - modules; C - periods; T - distance

Fundamentinių mokslų fakulteto Programų inžinerijos (612I30003) 2016-07-01 programos studijų rezultatų sąsajos su SDM rezultatais bei studijų ir studentų pasiekimų vertinimo metodais

Links of the Software Engineering (612I30003) of the Faculty of Fundamental Sciences with the course unit and evaluation methods of students achievements

Programos studijų rezultatai <i>Study programme outcomes</i>	SD(M) rezultatai <i>Course results</i>	Studijų metodai <i>Methods of studies</i>	Studento pasiekimų vertinimo metodai <i>Evaluation methods of student achievements</i>	Studentų pasiekimų vertinimo kriterijai pagal lygmenis <i>Assessments criteria of students achievements by Assessment levels</i>
Z2. Žinos ir galės paaiškinti algoritmų ir duomenų struktūras, programavimo paradigmas, kalbų, karkasų ir technologijų bei jų vystymosi tendencijas, programų kūrimo įrankius ir kodavimo standartus bei mokės tai paaiškinti ir taikyti. Z2. Consistent knowledge of algorithms and data structures, programming paradigms, languages and frameworks, programming technologies and their evolution tendencies, coding standards and software development tools, competency in its	Studentai įgyja žinias apie lygiagrečiųjų algoritmų sudarymą, lygiagretaus programavimo paradigmas ir įrankius: OpenMP, MPI, CUDA. Students acquire knowledge on the design of parallel algorithms, paradigms and tools of parallel programming: OpenMP, MPI, CUDA.	Teorinės paskaitos, laboratoriniai užsiėmimai, literatūros studijos, savarankiškas darbas. Theoretical lectures, laboratory works, study of literature, independent work.	Laboratorinis darbas, namų darbai, tarpinis egzaminas, egzaminas. Laboratory work, home works, intermediate exam, final exam.	Slenkstinis (5-6): studentas žino pagrindines dalyko sąvokas ir moka jomis naudotis sprendamas paprasčiausias užduotis. Tipinis (7-8): studentas žino pagrindines sąvokas, gali jas paaiškinti ir teorinius faktus geba tinkamai taikyti atlikdamas praktinių ir teorinių užduočių analizę, skaičiavimus taiko kompiuterines programas. Puikusias (9-10): studentas yra ne tik pasiekęs tipinį lygmenį, bet įgytas žinias geba

explanation and application.				taikyti naujose situacijose. Threshold (5-6): The student knows the basic concepts of the subject and is able to apply them solving simple problems. Typical (7-8): The student knows the basic concepts, is able to explain them and to apply theoretical facts correctly analyzing practical and theoretical problems, applies computer programs for calculations. Excellent (9-10): The student not only reached the typical level, but is able to apply the acquired knowledge in new situations.
SG3. Gebės taikyti skirtingas programavimo paradigmas, algoritmus, programinius karkasus, naudoti programų kūrimo įrankius, aplinkas ir priemones, testuoti programinę įrangą, kurti testavimo scenarijus pagal funkcinis ir nefunkcinis reikalavimus, vertinti testavimo rezultatus ir pateikti išvadas. SG3. Ability to apply different programming paradigms, algorithms, programming frameworks, to use relevant software development tools, environments and approaches, to apply software testing methods and develop test-cases according to functional and non-functional requirements, to evaluate testing results and make conclusions.	Studentai gebės taikyti skirtingus lygiagrečiųjų algoritmų sudarymo metodus, lygiagrečios programavimo paradigmas ir įrankius, testuoti ir vertinti lygiagrečiųjų algoritmų ir jų implementacijų efektyvumą. Students will be able to apply different methods for the design of parallel algorithms, use different paradigms and tools of parallel programming, test and evaluate the efficiency of parallel algorithms and their implementations.	Teorinės paskaitos, laboratoriniai užsiėmimai, literatūros studijos, savarankiškas darbas. Theoretical lectures, laboratory works, study of literature, independent work.	Laboratorinis darbas, namų darbai, tarpinis egzaminas, egzaminas. Laboratory work, home works, intermediate exam, final exam.	Slenkstinis (5-6): studentas žino pagrindines dalyko sąvokas ir moka jomis naudotis sprendžiamas paprasčiausias užduotis. Tipinis (7-8): studentas žino pagrindines sąvokas, gali jas paaiškinti ir teorinius faktus geba tinkamai taikyti atlikdamas praktinių ir teorinių užduočių analizę, skaičiavimams taiko kompiuterines programas. Puikūs (9-10): studentas yra ne tik pasiekęs tipinį lygmenį, bet įgytas žinias geba taikyti naujose situacijose. Threshold (5-6): The student knows the basic concepts of the subject and is able to apply them solving simple problems. Typical (7-8): The student knows the basic concepts, is able to explain them and to apply theoretical facts correctly analyzing practical and theoretical problems, applies computer programs for calculations. Excellent (9-10): The student not only

				reached the typical level, but is able to apply the acquired knowledge in new situations.
--	--	--	--	--

SD(M) sudarytojas (-ai) (parašas, vardas ir pavardė)

Course compiled by (full name, signature)

Vadimas Starikovičius

Katedros vedėjas (parašas, vardas ir pavardė)

Head of Department (full name, signature)

Raimondas Čiegis

SD(M) atestuojamas <i>The Course is certified</i>			
SD(M), skirtas studijų programai: <i>The Course for the programme of studies:</i>	Programų inžinerija		
SD(M) atestacija galioja: <i>Course certification is valid:</i>	nuo <i>from</i>		iki <i>till</i>

SD(M) atestavo <i>the Course certified by</i>	Fundamentinių mokslų fakulteto studijų komitetas		
Fakulteto studijų komiteto pirmininkas (vardas ir pavardė, parašas) <i>Chairman of the Studies committee (full name, signature)</i>		Data <i>Date</i>	