



MODULIO APRAŠAS

Modulio pavadinimas	Kodas
Loginis programavimas	

Dėstytojas	Padalinys
Koordinuojantis: prof. dr. Rimantas Vaicekuskas	Informatikos katedra Matematikos ir informatikos fakultetas Vilniaus universitetas
Kitas (-i):	

Studijų pakopa	Dalyko tipas
Pirmoji	Pasirenkamasis

Igyvendinimo forma	Vykdymo laikotarpis	Vykdymo kalbos
Auditorinė	5 ir 7 semestrai	Lietuvių, anglų

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai: Duomenų struktūros ir algoritmai, Matematinė logika.

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 loginio programavimo sąvokomis ir principais, ugdyti problemų sprendimo metodiką, naudojant Prolog instrumentines priemones.		
Bendrosios kompetencijos: - Bendravimas ir bendradarbiavimas (BK1). - Gebės savarankiškai efektyviai organizuoti savo darbą (BK1.3). - Nuolatinis mokymasis (BK2). - Gebės savarankiškai įsisavinti naujas žinias, metodus ir įrankius bei taikyti juos praktikoje (BK2.3).		
Dalykinės kompetencijos: - Konceptualių pagrindų žinios ir gebėjimai (DK4). - Gebės taikyti matematikos pagrindų, mokslo, inžinerijos, kompiuterių mokslo teorines žinias ir algoritminius principus programų sistemų kūrime (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 kompleksiskumą bei jų sprendimų būdų įgyvendinamumą (DK5.1).		
Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai
Suprasti deklaratyviosios specifikacijos, naudojančios Horno disjunktų logiką, principus.	Paskaitos, probleminis dėstymas, atvejų analizė, literatūros skaitymas, savarankiškas darbas, pavyzdžių analizė, konsultacijos, laboratoriniai darbai.	Laboratorinių darbų atlikimas bei rezultatų gynimas, egzaminas raštu (atvirojo, pusiau atvirojo bei uždarojo
Paašškinti Prolog programos procedūrinį veikimą.		
Naudoti Prolog sakinius ir struktūras, siekiant išreikšti žinių vaizdavimą ir loginį išvedimą.		

Kurti, derinti ir vertinti nedidelės ir vidutinės apimties Prolog programas kaip programinės įrangos prototipą.	tipo klausimai ir užduotys).
Naudoti išplėstines loginio programavimo technikas, sprendžiant problemas, kildinamas iš Dirbtinio intelekto (DI) uždavinių.	

Temos	Kontaktinio darbo valandos						Savarankiškų studijų laikas ir užduotys	
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai (LD)	Konsultavimas LD metu	Visas kontaktinis darbas	Savarankiškas darbas
Įvadas į loginio programavimo (LP) paradigmą. Deklaratyvaus ir procedūrinio programavimo akistata. Žinių raiška Horno disjunktai. Loginė programos semantika. LP ir DI sąsajos.	4				5	8	9	9
Prolog kaip loginio programavimo kalba. Procedūrinė Prolog semantika: tikslais-potiksliais grįstas vykdymo modelis, unifikacija (suvienodinimas) ir paieška su grįžimu (<i>back-tracking</i>).	4				5		9	9
Rekursyvusis programavimas Prologu: rekursyvių duomenų struktūrų – sąrašų, medžių- reprezentacija ir apdorojimas. Prolog aritmetikos priemonės.	2				4		6	6
Išplėstinės paieškos metodai: „generuotikrink“ paieškos šablonas, paieška būsenų erdvės grafe.	4				4		8	8
Simbolinių išraiškų apdorojimas, diferencijavimo pavyzdys.	2				2		4	4
Interpretatoriai ir žinių taisyklėmis besiremiančios sistemos.	4				4		8	8
Gramatikos ir kiti taikymai, susiję su natūraliosios kalbos supratimu.	3				2		5	5
Meta-predikatai <i>assert</i> , <i>retract</i> , <i>findall</i> . Paieškos į plotį panaudojimai. Paieška žaidimų medžiuose.	3				2		5	5
Neužbaigtos duomenų struktūros: skirtuminiai sąrašai, žodynai.	3				2		5	5
Prolog realizacijos. Lygiagretusis Prolog programų vykdymas.	3				2		5	3
Konsultacijos semestro eigoje		2					2	
Egzaminas (raštu)							2	
Total	32	2			32	8	68	62

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
Laboratoriniai darbai	40	4-rių užduočių atsiskaitymo laikas yra atitinkamai 3, 7, 11, 16 semestro savaitės.	Semestro metu studentas turi atlikti keturias užduotis (parašyti individualiai skirtas Prolog programas). 1-je užduotyje reikia apibrėžti Prolog predikatus, nusakančius loginį sąryšį tarp gerai žinomų objektų. 2-ji reikalauja rekursyvių apibrėžimų. 3-je užduotyje reikia apdoroti sąrašus. Paskutinioji (4 užduotis) turi išspręsti nurodytąją DI srities problemą. Kiekvienos užduoties sprendimas vertinamas iki 1 balo, ir sumoje galima surinkti 4 balus, atitinkančius 40% galutinio įvertinimo. Pateiktos užduotys vertinamos pagal tai, kokia apimtimi pilnai ir teisingai jos sprendžia duotąją problemą, ar studentas teisingai atsako į klausimus duotąją tema.
Egzaminas (raštu)	60	Egzaminų sesijos metu	Egzamino metu galima surinkti iki 6 taškų, kurie atitinka 60% galutinio įvertinimo. Egzaminas susideda iš trijų dalių. Pirmoje dalyje studentas turi pateikti atsakymus į skirtingus įvairaus sudėtingumo atvirojo, pusiau atvirojo bei uždarojo tipo klausimus (0-3 taškai). Antroje dalyje studentas turi pateikti praktinį pateiktos problemos sprendimą, kas apima ir Prolog kodo rašymą (0-2 taškai). Trečioje dalyje studentas turi pademonstruoti pateiktos temos suvokimą, parašydamas išbaigtą santrauką ir pateikdamas paaiškinamuosius pavyzdžius (0-1 taškai).

Reikalavimai dalyko vertinimui eksterno būdu	
Įvertinimas galimas eksterno būdu:	Netaikomas

Autorius	Leidimo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
Privalomoji literatūra				
Clocksin W.F., Mellish C.S.	2003	Programming in Prolog: Using the ISO Standard	5th ed.	Springer
Sterling L., Shapiro E.	1994	The Art of Prolog	2nd ed.	MIT Press
Papildoma literatūra				
Ulf Nilsson, Jan Maluszynski	2012	Logic, Programming and Prolog	2nd ed.	http://www.ida.liu.se/~ulfni/lpp/
Ivan Bratko	2000	Prolog Programming for Artificial Intelligence	3rd ed.	Addison Wesley



MODULE DESCRIPTION

Module title	Module code
Logic Programming	

Lecturer(s)	Department where the module is delivered
Coordinator: prof. dr. Rimantas Vaicekauskas Other lecturers:	Department of Computer Science 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	5 th and 7 th semester	Lithuanian and English

Prerequisites
Prerequisites: Data structures and algorithms, Mathematical logic.

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 introduce the key concepts and principles of the Logic programming, to present Prolog problem-solving techniques. Generic competences: - Communication and collaboration (GC1). - An ability to organise their own work independently (GC1.3). - Life-long learning (GC2) - 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). - 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).		
Learning outcomes of the module: students will be able to	Teaching and learning methods	Assessment methods
- Understand the principles of declarative specification using Horn clause logic. - Explain procedural behavior of the Prolog program. - Use Prolog statements and structures for knowledge representation and reasoning.	Lectures, problem-oriented teaching, case studies, literary reading, individual work, tutorials, laboratory work.	Laboratory works and results presentation, written exam (open, semi-open and close-

- Create, debug, and evaluate small-to-moderate size Prolog program as software prototype. - Use advanced Logic Programming techniques for solving problems related to the Artificial Intelligence (AI) applications.	ended questions and tasks).
--	-----------------------------

Content: breakdown of the topics	Contact hours						Self-study work: time and assignments		
	Lectures	Tutorials	Seminars	Practice	Laboratory work	Tutorial during LW	Contact hours	Self-study hours	Assignments
Introduction to logic programming (LP) paradigm. Declarative vs. procedural programming. Representation of knowledge with Horn clauses. The meaning of the logic program. Relation between LP and Artificial Intelligence.	4				5	8	9	9	Individual reading. Self-preparation for laboratory works. Self-control tasks.
Prolog as a logic programming language. Procedural semantics of Prolog: execution model in term of goals and sub-goals, unification (pattern matching) and backtracking	4				5		9	9	
Recursive programming with Prolog: representation and processing of simple data structures– lists and trees. Prolog arithmetic	2				4		6	6	
Advanced search techniques: generate –and – test search pattern, searching space–state graphs.	4				4		8	8	
Manipulating symbolic expressions. Symbolic differentiation example	2				2		4	4	
Interpreters and rule-based systems.	4				4		8	8	
Grammars and applications to natural language understanding.	3				2		5	5	
Meta-predicates <i>assert</i> , <i>retract</i> , <i>findall</i> . Breadth–first search applications. Searching games tree.	3				2		5	5	
Incomplete data structures: difference lists, dictionaries.	3				2		5	5	
Prolog implementations. Parallel Prolog.	3				2		5	3	
Tutorials during the semester		2					2		
Final exam (written)							2		
Total	32				32	8	68	62	

Assessment strategy	Weight %	Deadline	Assessment criteria
Laboratory works	40	There are 4 assignments with deadlines 3 th , 7 th , 11 th , 16 th week of the semester.	During the semester, a student is required to carry out 4 laboratory works (individual Prolog programming assignments). The 1 st laboratory work is to describe logical relations between well-known entities. The 2 nd assignment requires using recursive rules. The third one is about list processing. The 4 th is to write a Prolog program that solves an AI-related problem. Each laboratory work is evaluated from 0 to 1 points and total (4 points) is equivalent to 40% of the final score. Assessment criteria are: soundness and correctness of the program, relevance to the Prolog programming paradigm, ability to defend solutions and answer to the questions.

Exam (written)	60	During exam session	During the exam, it is possible to get at most 6 points, which are equivalent to 60% of the final score. The exam is divided into 3 parts. For the first part, the student must answer various questions (open, semi-open and close-ended questions and tasks) of diverse complexity (0-3 points). For the second part, the student must practically solve a given problem, which involves writing the code in Prolog (0-2 points). For the third part, the student must demonstrate an understanding of the given topic by writing a thorough summary and providing explanatory examples (0-1 points).
----------------	----	---------------------	---

Author	Publishing year	Title	Number or volume	Publisher or URL
Required reading				
Clocksin W.F., Mellish C.S.	2003	Programming in Prolog: Using the ISO Standard	5 th ed.	Springer
Sterling, L. and Shapiro, E.	1994	The Art of Prolog	2 nd ed.	MIT Press
Recommended reading				
Ulf Nilsson, Jan Maluszynski	2012	Logic, Programming and Prolog	2 nd ed.	http://www.ida.liu.se/~ulfni/lpp/
Ivan Bratko	2000	Prolog Programming for Artificial Intelligence	3 rd ed.	Addison Wesley