



## STUDIJŲ DALYKO (MODULIO) APRAŠAS

Dalyko (modulio) pavadinimas	Kodas
Dirbtinis intelektas: klasikiniai principai	

Anotacija
Kursas apima klasikinio dirbtinio intelekto (DI) pagrindus. Tai „geras senamadiškas DI“ (angl. <i>Good Old-Fashioned Artificial Intelligence</i> , GOFAI, žr. <a href="https://en.wikipedia.org/wiki/GOFAI">https://en.wikipedia.org/wiki/GOFAI</a> ). Kurse pabrėžiamas klasikinių DI metodų programavimas, pvz., paieška su grįžimais (angl. <i>backtrack</i> ). Aiškinamas uždavinių sprendimas ( <i>problem solving</i> ) intelektualizuotos euristinės paieškos būdais. Programuojama paieška eksponencialinio sudėtingumo erdvėje, kuri praktiškai neišsemiama. Kursas neapima mašininio mokymo(si) ir neuroninių tinklų. Tačiau trumpai pristatomos pastarojo dešimtmečio DI kryptys, pvz., tekstų generavimas su (pašnekesių) robotu (Chat)GPT ( <i>generative pre-trained transformer</i> ). Čia aktualios DI keliamos rizikos ir DI etika.

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

Studijų pakopa	Dalyko (modulio) tipas
Pirmoji	Pasirenkamas

Igyvendinimo forma	Vykdyto laikotarpis	Vykdyto kalba (-os)
Auditorinė	5 ir 7 semestrai	Lietuvių arba anglų

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai: Programavimo algoritmine kalba pagrindai, pvz., C, C#, Java, Pascal, Python ar kt.

Dalyko (modulio) apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5	130	64	66

Dalyko (modulio) tikslas: studijų programos ugdomos kompetencijos
<p>Modulio tikslas – 1) perprasti dirbtinio intelekto filosofinį klausimą „ką mašina gali“ ir 2) įgyti programavimo gebėjimus kurti intelektualizuotas sistemas, kuriose kompiuteris imituoja žmogaus veiksmus.</p> <p><b>Bendrosios kompetencijos:</b></p> <ul style="list-style-type: none"><li>Gebės raštu ir žodžiu perteikti informaciją, idėjas, problemas ir sprendimus valstybine ir užsienio kalba, bendraudamas su specialistais ir ne specialistais (BK1.1).</li><li>Gebės atlikti literatūros paiešką ir analizę, naudoti duomenų bazes ir kitus informacijos šaltinius (BK2.2).</li><li>Gebės savarankiškai įsisavinti naujas žinias, metodus ir įrankius bei taikyti juos praktikoje (BK2.3).</li></ul> <p><b>Dalykinės kompetencijos:</b></p> <ul style="list-style-type: none"><li>Supras pagrindines programų sistemų inžinerijos koncepcijas bei sąvokas, įskaitant kelias priešakines sritis, suvoks galimas taikymo sritis ir žinos disciplinos aprėptį (DK4.1).</li><li>Gebės taikyti matematikos pagrindų, mokslo, inžinerijos, kompiuterių mokslo teorines žinias ir algoritminius principus programų sistemų kūrime (DK4.2).</li><li>Gebės išvelgti naujas programų sistemų taikymo galimybes, įvertinti taikomosios srities žinių poreikį, problemų kompleksumą bei jų sprendimų būdų įgyvendinamumą (DK5.1).</li><li>Gebės projektuoti, įgyvendinti ir įvertinti programų sistemą, procesą, komponentą ar paslaugą, atitinkančią reikalavimus (DK5.3).</li><li>Gebės derinti teoriją ir praktiką programų sistemų taikymo įvairiose srityse uždavinių sprendimui, įvertinant technologinį, ekonominį, socialinį ir teisinį kontekstą (DK6.1).</li><li>Gebės parinkti ir panaudoti tinkamus šiuolaikinius metodus, modelius, problemų sprendimo šablonus, įgūdžius bei įrankius, būtinus programų sistemų kūrimui ir priežiūrai, įskaitant naujas taikymo sritis (DK6.2).</li></ul>

<ul style="list-style-type: none"> <li>Gebės panaudoti esamą kompiuterių techninę ir programinę įrangą, identifikuoti, perprasti ir taikyti perspektyvias technologijas (DK6.3).</li> <li>Gebės formuluoti racionalius, efektyvius kaštų ir laiko prasme problemų sprendimo variantus, remdamasis svarbiausiomis kaštų ir produktyvumo vertinimo žiniomis ir metodais (DK6.5).</li> </ul>		
Dalyko (modulio) studijų siekiniai	Studijų metodai	Vertinimo metodai
Suprasti kertines DI sąvokas: uždavinių sprendimas paieškos būdais, valdymas su grįžimais, euristika, sprendėjas, planuotojas, agentas ir kt.	Probleminis dėstymas, mokslinės literatūros nagrinėjimas individualiai, programavimas.	Egzaminas (raštu), namų darbų vertinimas. Nesudėtingi namų darbai vertinami atlikta/neatlikta, o sudėtingesni – balu. Kriterijai: programavimo kokybė, atsiskaitymas laiku, problematikos supratimo įrodymų pateikimas.
Įgyti programavimo dirbtiniame intelekto gebėjimus tokius kaip tiesioginis ir atbulinis išvedimas su produkcinėmis taisyklėmis.		
Suprasti DI esmę pagal Tiuringo testą – ką gali ir ko negali kompiuteris – ir formaliosios logikos peržengimą priimant sprendimą, kai argumentai balansuojami.		

Temos	Kontaktinio darbo valandos						Savarankiškų studijų laikas ir užduotys		
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai	Praktika	Visas kontaktinis darbas	Savarankiškas darbas	Užduotys
1. Dirbtinio intelekto istorija. DI „burbulas“ ( <i>hype</i> ). Tiuringo testas ir žodžių „mašina“, „gali“ bei „mąstyti“ aiškinimas; žr. taip pat <a href="https://en.wikipedia.org/wiki/Turing_test">https://en.wikipedia.org/wiki/Turing_test</a> . Galvosūkiškai kaip intelekto reikalaujanti veikla.	2			2			4	4	Hanojaus bokštų programa. <i>Literatūra</i> : paskaitų konspektas (Čyras 2021), (Nilsson 2010), (Luger 2009),
2. Dirbtinio intelekto produkcijų sistema: 1) globali duomenų bazė (būsenų erdvė), 2) produkcijų aibė (veiksmai) ir 3) valdymo sistema (tikslų būsenos testas). Algoritmas PRODUCTION. Pavyzdžiai: šachmatų lentos apėjimas žirgo ėjimu ir kt.	2			2			4	4	Žirgo maršruto programa. <i>Literatūra</i> : konspektas, (Nilsson 1982; 1998)
3. Valdymo su grįžimais algoritmas BACKTRACK. Pavyzdžiai: 8 valdovių uždavinys ir kt.	2			2			4	4	<i>Literatūra</i> : konspektas, (Nilsson 1982; 1998)
4. Modifikuotas algoritmas BACKTRACK1 su atmintimi tikslu išvengti užsiklindimo. Paieška į gylį labirinte.	2			2			4	4	Programa kelio paieškai labirinte į gylį. <i>Literatūra</i> : konspektas; (Nilsson 1982)
5. Euristikos samprata. Pavyzdžiai N valdovių uždavinyje.	2			2			4	4	<i>Literatūra</i> : konspektas, (Nilsson 1982; 1989)
6. Paieška į plotį (bangos metodas). Kompiuterinių agentų tipai pagal intelekto laipsnį: refleksyvus, racionalus ir kt.	2			2			4	4	Programa kelio paieškai labirinte į plotį. <i>Literatūra</i> : konspektas, (Russell & Norvig 2020), (Nilsson 1998)
7. Algoritmas GRAPHSEARCH. Sprendėjo ir planuotojo skirtumai. Algoritmas A*.	2			2			4	4	<i>Literatūra</i> : konspektas, (Nilsson 1982; 1998)
8. Tiesioginis išvedimas ( <i>forward chaining</i> , FC, nuo faktų prie tikslo, be rekursijos) ir atbulinis išvedimas ( <i>backward chaining</i> , BC, nuo tikslo prie faktų, rekursinis) su taisyklėmis $A_1, \dots, A_n \rightarrow B$ .	2			4			6	4	Programa tiesioginiam išvedimui (FC). <i>Literatūra</i> : konspektas, (Negnevitsky 2011)
9. Lipimo į kalną ( <i>hill climbing</i> ) strategija paieškai.	2			0			2	4	<i>Literatūra</i> : konspektas, (Nilsson 1982; 1998)

10. Rezoliucijų metodas. Dedukcija ir <i>modus ponens</i> taisyklė. Žiniomis grindžiamas samprotavimas. Logika grindžiamas žinių vaizdavimas.	2		2		4	6	Programa atbuliniam išvedimui (BC). <i>Literatūra:</i> konspektas, (Nilsson 1998), (Russell & Norvig 2020)
11. Ekspertinių sistemų architektūros elementai: faktai, taisyklės ir išvedimo mašina. Pavyzdys.	2		2		4	4	<i>Literatūra:</i> konspektas, (Nilsson 1998)
12. Samprotavimas pagal keletą kriterijų ar argumentų, pvz., „geras, bet brangus“ ar „prastas, bet pigus“. Pavyzdys teisėje: nesuderinamų tikslų kompromisas kaip balansas pasveriant galimus sprendimus.	2		2		4	4	<i>Literatūra:</i> konspektas; (Bench-Capon & Prakken 2006)
13. Apsipirkimui internete skirtos programos esmė (žr. Russell & Norvig 2003, p. 344–348). Prekių kategorijų medis kaip ontologija.	2		2		4	4	<i>Literatūra:</i> konspektas, (Russell & Norvig 2003; 2020)
14. Žinių vaizdavimo metodai: struktūrinis vaizdavimas, freimai ir objektai, semantiniai tinklai.	2		2		4	4	<i>Literatūra:</i> konspektas, (Nilsson 1998)
15. Ekstensionalinė ir intensionalinė reliacinės struktūros, konceptualizacija ir ontologija.	2		2		4	4	<i>Literatūra:</i> <a href="https://www.researchgate.net/publication/226279556_What_Is_an_Ontology">https://www.researchgate.net/publication/226279556_What_Is_an_Ontology</a>
16. Aktualios DI kryptys. Tekstų generavimas su GPT (generative pre-trained transformer), DI etika, DI „burbulo“ ( <i>hype</i> ) kritika. Egzamino klausimų aptarimas.	2		2		4	4	<i>Literatūra:</i> <a href="https://arxiv.org/abs/2303.12712">https://arxiv.org/abs/2303.12712</a> . Pasirengimas egzaminui.
<b>Iš viso</b>	<b>32</b>		<b>32</b>		<b>64</b>	<b>66</b>	

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
1. Keletas paprastų programos, kurių algoritmai aptariami per paskaitas. Pavyzdžiui, Hanojaus bokštai, žirgo maršrutas šachmatų lentoje, kelio paieška labirinte (į gylį, į plotį), A*. <i>Tikslas:</i> dirbtinio intelekto požiūriai (sprendėjo ir planuotojo) į rekursines programas, kurios gali būti žinomos iš kitų kursų.	0%	Dėstytojo nurodytais terminais; kiekviena programa per 1–2 savaites. Vėliausiai – septintą savaitę	Užduotys atliekamos kaip namų darbai ir vertinamos „atlikta“ ar „neatlikta“. Atsižvelgiama į programavimo kultūrą: uždavinio aprašymas ir įvestis–išvestis. Testų dokumentavimas: 1) įvesties duomenų pakartojimas ekrane, 2) struktūrizuotas vykdymo protokolas ( <i>log, trace</i> ), 3) rezultato išvestis. Sankcija už vėlavimą: egzamino balas mažinamas (subjektyviai, bet protinga apimtimi). Neatsiskaitę visų užduočių neprileidžiami egzaminui. Rekomenduojama lankyti bent 75% paskaitų ir seminarų, nes kitaip vargiai įmanoma suprasti esmę ir pasirengti egzaminui.
2. Programos tiesioginiam išvedimui (FC) ir atbuliniam išvedimui (BC).	30%	11 savaitė FC, 15 savaitė BC	Vertinama pažymiu. Algoritmai ir testai nagrinėjami per paskaitas ir pratybas. Programos atsiskaitymui 4 savaitės: pirma savaitė – programa įveda pradinis duomenis ir juos spausdina, antra – dirba su paprastais testais, trečia – sudėtingesni testai, ketvirta – atsiskaitymas. Atkreipti dėmesį į programavimo ir dokumentavimo kultūrą: taisyklinga kalba, išsamus aprašymas, protinga apimtis, darbas atliktas savarankiškai, nėra plagijavimo, pateikiami įrodymai, kad problematika suprasta ir literatūra skaityta, o ne cituota aklaiai.
3. Egzaminas raštu.	70%		Du klausimai: 1) teorija ir 2) pratimas. Pratimo pavyzdys: duotam grafiui nubraižyti paieškos medžius keliui nuo viršūnės <i>s</i> iki <i>t</i> rasti procedūromis BACKTRACK1 ir GRAPHSEARCH „į gylį“ bei nurodyti skirtumus. Kaupiamasis balas nenusveria studento naudai nepatenkinamo atsakymo „4“ į egzamino klausimą (pvz., „nieko nežinau“). Kitaip sakant, teigiamam (≥5) egzamino pažymiu būtina atsakyti teigiamai ir į teoriją, ir į pratimą.

Reikalavimai dalyko vertinimui eksterno būdu	
Įvertinimas galimas eksterno būdu:	Taip
VU MIF studentai gali kartoti kursą eksterno būdu. Būtina atlikti visas pratybų užduotis (1. keletas sprint-užduočių programų; 2. tiesioginio išvedimo programa; 3. atbulinio išvedimo programa). Paskui laikyti egzaminą. Į ankščiau klausytų kursų įvertinimus nėra atsižvelgiama.	

Autorius	Leidimo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
<b>Privaloma literatūra</b>				
1. Vytautas ČYRAS	2022	Intelektualios sistemos (paskaitų konspektas)		<a href="https://klevas.mif.vu.lt/~cyras/AI/konspektas-intelektualios-sistemos.pdf">https://klevas.mif.vu.lt/~cyras/AI/konspektas-intelektualios-sistemos.pdf</a>
2. Nils NILSSON	1997	Artificial intelligence: a new synthesis		Elsevier Science & Technology. Yra VU MIF skaitykloje, šifras 004.8 Ni-133. <a href="https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=1179844">https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=1179844</a>
3. Stuart RUSSELL, Peter NORVIG	2020	Artificial intelligence: a modern approach (4th edition)		Prentice Hall, 1115 p. Yra VU MIF, 2nd edition (2003), šifras 007 Ru122. <a href="https://aima.cs.berkeley.edu">https://aima.cs.berkeley.edu</a> . <a href="https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5495854">https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5495854</a>
4. Michael NEGNEVITSKY	2011	Artificial intelligence: a guide to intelligent systems (3rd edition)		Pearson Education. Yra VU MIF 2nd edition (2005), šifras 004.8/Ne-44. <a href="https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5186211">https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5186211</a>
<b>Papildoma literatūra</b>				
1. Stanislovas NORGĖLA	2007	Logika ir dirbtinis intelektas		TEV, Vilnius. Yra VU MIF, šifras 16 No66
2. Nils NILSSON	2009	The quest for artificial intelligence: a history of ideas and achievements		Cambridge University Press. Yra VU MIF, šifras 004.8 Ni-133. <a href="https://doi.org/10.1017/CBO9780511819346">https://doi.org/10.1017/CBO9780511819346</a>
3. Nils NILSSON	1982	Principles of artificial intelligence		Springer-Verlag
4. George LUGER	2009	Artificial intelligence: structures and strategies for complex problem solving (6th ed.)		Addison-Wesley, 928 p. <a href="https://www.cs.unm.edu/~luger/">https://www.cs.unm.edu/~luger/</a> . Yra VU MIF, šifras 004.8/Lu-59

Atnaujinta 2023.04.23



## COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Artificial Intelligence: Classical Principles	20477

Summary
<p>This course comprises only elements of artificial intelligence (AI) and not the whole variety of AI techniques. Emphasis is on the classical AI, a.k.a. Good Old-Fashioned Artificial Intelligence, GOFAI, <a href="https://en.wikipedia.org/wiki/GOFAI">https://en.wikipedia.org/wiki/GOFAI</a>. Hence, the course comprises the principles of AI, search methods, and logic-based representations. Machine learning and artificial neural networks are out of scope of this one-semester course.</p>

Lecturer(s)	Department
<p><b>Coordinator:</b> assoc. prof. Vytautas ČYRAS</p> <p><b>Other lecturers:</b> -</p>	<p>Department of Software Engineering Institute of Computer Science Faculty of Mathematics and Informatics Vilnius University</p>

Cycle	Type of the course unit
1 <sup>st</sup> (BA)	Non-compulsory

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	5, 6, 7 semester	English

Prerequisites
<p>Programming skills in a programming language, for example, C, C#, Java, Pascal, Perl, Python, etc.</p>

Number of credits allocated	Student's workload	Contact hours	Individual work
5	130	64	66

Purpose of the course unit: programme competences to be developed		
<p>To develop competencies and skills in artificial intelligence: to understand the philosophy of AI and the question “Can machines think?”. To gain programming skills at creating future intelligent systems, which extend the limits of machine intelligence of the past and use the elements of human intelligence in problem solving.</p> <p><b>Generic competences:</b> communication and collaboration, life-long learning, and social responsibility.</p> <p><b>Specific competences:</b> knowledge and skills of underlying conceptual basis, software development knowledge and skills, technological and methodological knowledge and skills, and professional competence.</p>		
Learning outcomes of the course unit: students will be able	Teaching and learning methods	Assessment methods
<p>To understand the fundamental concepts of AI: problem solving by search, backtracking, solver, planner, software agent, the Turing test, etc.</p> <p>To program basic techniques of inference: backtrack (depth-first), breadth-first search, forward chaining and backward chaining.</p> <p>To understand the essence of AI: differentiating between human intelligence and machine intelligence, extra-logical choice in decision making, e.g., in legal reasoning.</p>	<p>Problem-based teaching, individual reading, writing programs.</p>	<p>Written examination, assignments.</p> <p>Criteria: the quality of programming; accomplishing the assignments in due time – no delay; understanding the subject matter of AI.</p>

Course content: breakdown of the topics	Contact hours						Individual work: time and assignments		
	Lectures	Tutorials	Seminars	Practice	Laboratory work (LW)	Praktika	Contact hours	Individual work	Assignments
1. Artificial intelligence (AI): a history, a philosophy, and the hype. Examples of intelligent tasks. AI as a discipline within a classification of computing. The Turing test and interpreting the question “Can machines think?”	2			2			4	4	The Tower of Hanoi puzzle. <i>Literature</i> : the course-book (text-book) (Čyras 2021), (Nilsson 2010), (Luger 2009), <a href="https://en.wikipedia.org/wiki/Turing_test">https://en.wikipedia.org/wiki/Turing_test</a>
2. A system of artificial intelligence: 1) a global data base, 2) a set of production rules, and 3) a control system. Procedure PRODUCTION. Examples: the Tower of Hanoi problem, etc.	2			2			4	4	The Knight’s tour program. <i>Literature</i> : textbook, (Nilsson 1982; 1998)
3. Procedure BACKTRACK. A problem solving example: the Knight’s tour problem.	2			2			4	4	<i>Literature</i> : textbook, (Nilsson 1982; 1998)
4. Avoiding loops in labyrinth depth-first search (DFS) with BACKTRACK1, a modified procedure.	2			2			4	4	Programming DFS in a labyrinth. <i>Literature</i> : textbook, (Nilsson 1982)
5. The concept of heuristic. Examples in the N-queens problem.	2			2			4	4	<i>Literature</i> : textbook, (Nilsson 1982; 1989)
6. Breadth-first search (BFS). The lists OPEN and CLOSED and the shortest path. The types of intelligent agents: reflexive and rational agents.	2			2			4	4	Programming BFS in a labyrinth. <i>Literature</i> : textbook, (Russell & Norvig 2020), (Nilsson 1998)
7. Procedure GRAPHSEARCH. The solver and the planner. A* search algorithm.	2			2			4	4	<i>Literature</i> : textbook, (Nilsson 1982; 1998)
8. Forward chaining (FC): from the facts to the goal, non-recursive. Backward chaining (BC): from the goal to the facts, recursive. Rule format: $A_1, \dots, A_n \rightarrow B$ .	2			4			6	4	Programming FC. <i>Literature</i> : textbook, (Negnevitsky 2011)
9. Hill climbing strategy.	2			0			2	4	<i>Literature</i> : textbook, (Nilsson 1982; 1998)
10. Knowledge-based reasoning, the resolution rule, deduction. Inference forwards and backwards. Logic-based knowledge representation.	2			2			4	6	Programming BC. <i>Literature</i> : textbook, (Nilsson 1998), (Russell & Norvig 2020)
11. Elements of expert systems architecture: a knowledge base (facts and rules), an inference engine and user interface.	2			2			4	4	<i>Literature</i> : textbook, (Nilsson 1998)
12. Extra-logical choice in decision making, e.g., “low-quality but cheap” versus “good-quality but expensive”. Transforming the problem of impossibility of achieving several goals into a weighing problem. Deduction and abduction rules. Defeating argumentation trees.	2			2			4	4	<i>Literature</i> : textbook, (Bench-Capon & Prakken 2006)
13. The Internet shopping world (see Russell & Norvig 2003, p. 344–348): specifying a search engine. A category tree as an ontology.	2			2			4	4	<i>Literature</i> : textbook, (Russell & Norvig 2003; 2020)
14. Knowledge representation methods: structural representation, frames, semantic networks.	2			2			4	4	<i>Literature</i> : textbook, (Nilsson 1998)

15. Extensional relational structure, intensional relational structure (conceptualization), intended models, and ontology.	2		2		4	4	<i>Literature:</i> <a href="https://www.researchgate.net/publication/226279556_What_Is_an_Ontology">https://www.researchgate.net/publication/226279556_What_Is_an_Ontology</a>
16. Summing up the principles of artificial intelligence. Discussing the examination.	2		2		4	4	Preparing to the exam
<b>Total</b>	<b>32</b>		<b>32</b>		<b>64</b>	<b>66</b>	

Assessment strategy	Weight	Deadline	Assessment criteria
1. Simple programs. For instance, the Tower of Hanoi puzzle, the Knight's tour, labyrinth path search (DFS, BFS), A*.	0%	Week 6	Each assignment shall be finished in due time, no delay. Forward chaining and backward chaining programs shall be finished in four weeks. Week 1: your program reads initial data and prints it. Week 2: simple tests. Week 3: complex testing. Week 4: – acceptance.
2. Forward and backward chaining programs (FC and BC).	30%	Week 11 FC, Week 15 BC	<p>Assignments' assessment criteria: the quality of programming, testing and documenting. The output of a test comprises 1) input data from a file, 2) execution trace (log), and 3) the results. The code shall contain comments, step numbering, and explanations of data structures.</p> <p><b>A review instead of FC/BC.</b> An option (e.g., for students who are less-skilled in programming) is (1) investigating a chosen problem, article or book, (2) writing a review, and (3) making a presentation. Font 12 pt, spacing 1, ~20 pages with references. Evidence of understanding the problem.</p> <p><b>For examination</b> it is obligatory to pass all assignments. Assignments rate 30% of the exam's score.</p> <p>It is strongly recommended to attend <math>\geq 75\%</math> of lectures and practice.</p>
3. Examination.	70%	Examination date	<p>The examination comprises a theory question and an exercise. The exercise: for a given graph, draw search trees from a node <math>s</math> to <math>t</math> according to procedures BACKTRACK1 and GRAPHSEARCH “depth-first”; indicate distinctions.</p> <p>The assignments score counts if each question is answered in the affirmative. In other words, the exam grade is placed positive only if each exam question is answered in the affirmative (<math>\geq 5</math> out of 10).</p>

Author	Publishing year	Title	Number or volume	Publisher or URL
<b>Required reading</b>				
1. Vytautas ČYRAS	2023	Artificial Intelligence (a textbook)		<a href="https://klevas.mif.vu.lt/~cyras/AI/ai-cyras.pdf">https://klevas.mif.vu.lt/~cyras/AI/ai-cyras.pdf</a>
2. Nils NILSSON	1997	Artificial intelligence: a new synthesis		Elsevier Science & Technology. VU MIF, 004.8 Ni-133. <a href="https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=1179844">https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=1179844</a>
3. Stuart RUSSELL, Peter NORVIG	2020	Artificial intelligence: a modern approach (4th edition)		Prentice Hall, 1115 p. VU MIF, 2nd edn (2003), Ru122. <a href="https://aima.cs.berkeley.edu">https://aima.cs.berkeley.edu</a> . <a href="https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5495854">https://ebookcentral.proquest.com/lib/viluniv-ebooks/detail.action?docID=5495854</a>
4. Michael NEGNEVITS-KY	2011	Artificial intelligence: a guide to intelligent systems (3rd edition)		Pearson Education. VU MIF, 2nd edn. (2005), 004.8/Ne-44. <a href="https://ebookcentral.proquest.com/lib/viluniv-">https://ebookcentral.proquest.com/lib/viluniv-</a>

				<a href="https://doi.org/10.1017/CBO9780511819346">ebooks/detail.action?docID=5186211</a>
<b>Recommended reading</b>				
1. Nils NILSSON	2009	The quest for artificial intelligence: a history of ideas and achievements		Cambridge University Press. VU MIF, 004.8 Ni-133. <a href="https://doi.org/10.1017/CBO9780511819346">https://doi.org/10.1017/CBO9780511819346</a>
2. Nils NILSSON	1982	Principles of artificial intelligence		Springer-Verlag
3. George LUGER	2009	Artificial intelligence: structures and strategies for complex problem solving (6th ed.)		Addison-Wesley, 928 p. <a href="https://www.cs.unm.edu/~luger/">https://www.cs.unm.edu/~luger/</a> . VU MIF, 004.8/Lu-59

Updated 11 July 2023