MODULE DESCRIPTION

Module title	Module code
Knowledge Representation	

Lecturer(s)	Department
Coordinator: assoc. prof. dr. Vytautas Čyras **Other lecturers: -**	Department of Software Engineering 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, 6, 7 semester	Lithuanian

Prerequisites
Prerequisites: Procedural Programming; Object Oriented Programming.

Number of credits allocated	Student's workload	Contact hours	Self-study hours
5	130	69	61

Purpose of the module: programme competences to be developed

Purpose of the module – develop competences and skills in knowledge representation (KR): ways to provide declarative knowledge in machine-readable form, the resource description of the Semantic Web and the limits of decidability.

Generic competences:

- Communication and collaboration *(GK1)*.

Specific competences:

- Knowledge and skills of underlying conceptual basis *(SK4)*:
- Software development knowledge and skills *(SK5)*.
- Technological and methodological knowledge and skills, professional competence *(SK6)*.

Learning outcomes of the module: students will be able to	Teaching and learning methods	Assessment methods
Understand the fundamental concepts of knowledge representation (KR) in the sense of artificial intelligence and the relevance of KR to the Semantic Web.	Problem-based teaching, individual reading, writing programs.	Written examination, assignments (laboratory works) Criteria: quality of programs; writing a report; understanding knowledge representation.
Understand deductive and abductive reasoning, to program basic inference techniques (forward and backward) and the elements of multiagent systems.		
Understand the essence of knowledge representation problem, differentiating between human intelligence and machine intelligence according to the Turing test and the infeasibility of representing wisdom.		

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**	**Assignments**
1. Knowledge representation and entailment. History of artificial intelligence. Knowledge representation as a branch or artificial intelligence and its relevance to the Semantic Web.	2			2			**6**	**3**	
2. Data, information, knowledge and wisdom. The Turing test. Searle's "Chinese Room" thought experiment. Expanding boundaries between human and machine intelligence while developing knowledge-based systems.	2			2			**4**	**3**	
3. Procedural and declarative knowledge representation. Input-output of the sequential composition of programs. Program synthesis.	2			2			**4**	**2**	Individual reading. Programming individually (a simple program).
4. Knowledge representation methods: 1) logical KR – propositional logic and predicate logic; 2) procedural KR – production rules and inference engine control system; 3) network KR – semantic networks and conceptual graphs; 4) structured KR – frames and objects.	2						**2**	**2**	
5. Knowledge representation and knowledge management. Logic in reasoning. Deduction and the *modus ponens* rule.	2			2			**4**	**2**	
6. Semantic networks. General and individual concepts. Interpretation of the relationships *is-a* and *instance-of*.	2			2			**4**	**2**	
7. Semantic networks according to Russell & Norvig (2003). Inheritance. Inverse relationship. Reification.	2					*8*	**2**	**2**	
8. Conceptual models in database management systems. Extensional and intensional relations and a conceptualization. Different nature of knowledge representation and knowledge visualization.	2			2			**4**	**2**	Individual reading. Programming individually (an elaborated program)
9. Problem solving by search. The BACKTRACK procedure. Heuristics. The GRAPHSEARCH procedure. Solvers and planners.	2			2			**4**	**4**	
10. Forward chaining (from facts to goal, non-recursive, FC) and backward chaining (from goal to facts, recursive) with rules (format A1,…, An → B).	2			16			**18**	**11**	
11. Distinguishing knowledge sources. Elements of expert systems architecture: facts, rules, and inference engine.	2			2			**4**	**4**	
12. Infeasibility of representing wisdom. Extralogical choice in decision making, e.g., "low-quality but cheap" versus "good-quality but expensive". Transforming the problem of infeasibility of achieving several goals into a weighing problem. Abduction and deduction rules. Argumentation trees and defeasible reasoning.	2						**2**	**2**	Individual reading. Programming individually (a complicated program)

13. Representing the categories of things. The Internet shopping world (Russell & Norvig 2003, p. 344-348).	2						**2**	**2**	
14. An extensional relational structure, a world, an intensional relational structure, a conceptualization and ontology.	2						**2**	**2**	
15. Distinguishing between the presentation of Web pages and annotating Web resources. A need to share knowledge. An introduction to Semantic Web languages: graphs, RDF and XML.	2						**2**	**2**	
16. Discussing examination exercise and questions	2						**2**		
17. Consultation		2				*2*	**2**		
18. Examination							**3**	**16**	
Total	**32**	**2**		**32**		*8*	**69**	**61**	

Assessment strategy	**Weight %**	**Deadline**	**Assessment criteria**
1. A simple program (simple agent actions)	5%	Week 5	The course is designed to students who have or have not fundamentals in artificial intelligence. The latter are recommended writing forward chaining (FC) and backward chaining (BC) programs in order to program the elements of inference and to understand the differences. Students who have fundamentals in artificial intelligence are recommended to write a multi-agent program in AgentSpeak (Java extension) using Jason. Instead this, a student can also choose a topic he is interested in which is relevant to AI such as a survey, an heuristics, a computer game, a multi-agent application, etc. Assessment criteria: the quality of programming, documentation, problem description, pseudo-code, input-output, program's flow-diagram or UML diagrams, the (core) code, print data which comprises 1) input data from file, 2) execution trace (log), and 3) the resulting path. The code is required to relate the pseudo-code (the same labels shall be used). A detailed description of test cases, semantic networks, a core program with comments, numbered steps, data structures and their explanation. Reasonable volume (including annexes) and references. Minimum 10 pages in 12 pt font, spacing 1, proper citation. Evidence shows that the problem is understood and the work has been carried out on your own. Reading literature in the original language is mandatory. Each assignment (lab work) shall be done in due time, within 4-5 weeks. The first week – the program reads initial data and prints, the second week – simple examples, the third – complex examples, the fourth – draft report, the fifth – final report. For examination it is obligatory to pass the assignments. Assignments rate 30% the exam's score. It is recommended to attend 75% of lectures and exercises. Precedents have shown that not attending can cause difficulties in understanding the subject matter.
2. A more elaborated program (sequences of agent actions, inference elements)	10%	Week 10	
3. A complicated program (agents cooperate to achieve the goal, abductive reasoning elements, writing a program as a service in the Web)	15%	Week 15	
4. Examination	70%	Examination	The examination comprises a theory question and an exercise. The assignments' score counts if each exam question is answered in the affirmative (i.e. satisfactory or better, ≥ 5 out of 10 points). In other words, the score does not outweigh the student's unsatisfactory response (e.g., "I do not know") to an exam question, i.e. final exam grade is placed positive only if each exam question is answered in the affirmative.

Author	Publishing year	Title	Number or volume	Publisher or URL
Required reading				
Vytautas ČYRAS	2013	Intelligent Systems [coursebook in Lithuanian]		http://www.mif.vu.lt/~cyras/AI/konspektas-intelektualios-sistemos.pdf
Ronald BRACHMAN, Hector LEVESQUE	2004	Knowledge Representation and Reasoning		The Morgan Kaufmann Series in Artificial Intelligence, 381 p. VU MIF: 004.8/Br-04
George LUGER	2005	Artificial Intelligence: Structures and Strategies for Complex Problem Solving (fifth ed.)		Addison-Wesley, 928 p. http://www.cs.unm.edu/~luger/. VU MIF: 004.8/Lu-59
Stuart RUSSELL, Peter NORVIG	2003, 2010	Artificial Intelligence: A Modern Approach (2nd or 3rd ed.)		Prentice Hall, 1132 p. VU MIF (2nd ed.): 007 Ru122. http://aima.cs.berkeley.edu
Grigoris ANTONIOU, Paul GROTH, Frank van HARMELEN, Rinke HOEKSTRA	2012	A Semantic Web Primer (third ed.)		The MIT Press, 270 p. First edition (2004) VU MIF 004.8/An-154
Recommended reading				
Stanislovas NORGĖLA	2007	Logic and Artificial Intelligence [in Lithuanian]		TEV, Vilnius. MIF: 16 No66
Nils NILSSON	1998	Artificial Intelligence: A New Synthesis		Morgan Kaufmann Publishers. VU MIF 004.8/Ni-133
Michael NEGNEVITSKY	2005	Artificial Intelligence: A Guide to Intelligent Systems (2nd ed.)		Addison-Wesley. VU MIF: 004.8/Ne-44
Rafael BORDINI, Jomi Fred HÜBNER, Michael WOOLDRIDGE	2007	Programming Multi-agent Systems in AgentSpeak Using Jason		John Wiley, 273 p. VU MIF: 0/Bo-271
Mark STEFIK	1995	Introduction to knowledge systems		Morgan Kaufmann Publishers, 871 p. VU MIF: 0/St-137.