

SYLLABUS / FIȘA DISCIPLINEI
1. Information on the study programme / Date despre programul de studii

1.1. Institution / Instituția de învățământ superior	Universitatea de Vest din Timișoara
1.2. Faculty / Facultatea	Matematică și Informatică
1.3. Department / Departamentul	Computer Science (Informatică)
1.4. Study program field	Computer Science (Informatică)
1.5. Study cycle/ Ciclul de studii	MSc / master
1.6. Study programme / Programul de studii / calificarea*	Artificial Intelligence and Distributed Computing

2. Information on the course / Date despre disciplină

2.1. Title of the course / Denumirea disciplinei		Algorithm Synthesis and Mathematical Theory Exploration					
2.2. Teacher in charge of the course / Titularul activităților de curs		Lect. Dr. Isabela Drămnesc					
2.3. Teacher in charge of the seminar / Titularul activităților de seminar		Lect. Dr. Isabela Drămnesc					
2.4. Study year / Anul de studii	2	2.5. Semester / Semestrul	1	2.6. Examination type / Tipul de evaluare: E(xam)/C(olloquim)	C	2.7. Course type / Regimul disciplinei: M(andatory)/ E(lective)/ F(acultative)	E

3. Estimated study time (number of hours per semester) /Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Attendance hours per week / Număr de ore pe săptămână	3	out of which din care: 3.2 lecture/ curs	2	3.3. seminar/laborator	1
3.4. Attendance hours per semester / Total ore din planul de învățământ	42	out of which: 3.5 lecture / curs	28	3.6. seminar/laborator	14
Distribution of the allocated amount of time / Distribuția fondului de timp*					hours/ ore
Individual study /Studiu după manual, suport de curs, bibliografie și notițe					28
Supplementary documentation at library or using electronic repositories / Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate					14
Preparing for laboratories, homework, reports etc. /Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					38
Exams / Examinări					4
Tutoring / Tutorat					14
3.7. Total number of hours of individual study / Total ore studiu individual	98				
3.8. Total number of hours per semester / Total ore pe semestru	140				

3.9. Number of credits (ECTS) /5 Număr de credite	
--	--

4. Prerequisites (if it is the case) / Precondiții (acolo unde e cazul)

4.1. curriculum / de curriculum	Logic, Programming, Algorithmics, Automated Theorem Proving
4.2. skills / de competențe	Programming, capacity for individual and team work.

5. Requirements (if it is the case) / Condiții (acolo unde e cazul)

5.1. for the lecture / de desfășurare a cursului	Whiteboard, projector, laptop, Meet, Classroom, Forms, Remote connection
5.2. for the seminar, laboratory / de desfășurare a seminarului/laboratorului	Whiteboard, projector, laptop, Meet, Classroom, Forms, Remote connection

6. Acquired skills / Competențe specifice acumulate

Professional skills / Competențe profesionale	<p>Cognitive:</p> <ol style="list-style-type: none"> 1. understand the principles of Mathematical Theory Exploration and Algorithm Synthesis; 2. acquire the skills for exploring systematically mathematical theories; 3. study the main algorithm synthesis methods; 4. study the main proving method in first order predicate logic; 5. acquire the skills for using Mathematical Logic in Mathematics and Computer Science; 6. study the applications of Mathematical Logic to algorithm development and verification. <p>Technical: Perform concrete tasks with (1)-(6) above.</p> <p>Affective cognitive: Understand that the cognitive and technical skills above as important for a computer scientist.</p>
Transversal skills / Competențe transversale	Identify situations in which computational logic techniques and algorithms can play a role in problem solving and apply it.

7. Objectives of the course / Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. General objective / Obiectivul general al disciplinei	Understanding the principles of Mathematical Theory Exploration, acquire the skills for exploring systematically mathematical theories, study the main algorithm synthesis methods, study the main proving method in first order predicate logic, acquire the skills for using Mathematical Logic in Mathematics and Computer Science, study the applications of Mathematical Logic to algorithm development and verification.
--	--

7.2. Specific objectives / Obiectivele specifice	<p>Knowledge: Describe syntax of predicate logic, methods for systematically exploration of mathematical theories and for algorithm synthesis, tools.</p> <p>Abilities: Perform exercises and programming tasks with the notions introduced.</p> <p>Aptitudes: Identify situations where techniques from mathematical logic can be employed.</p>
--	---

8. Content / Conținuturi*

8.1. Lecture / Curs	Teaching strategies / Metode de predare	Remarks, details / Observații
1. Introduction. First order Predicate Logic. Syntax. Sequent calculus. Methods of proving	Q&A/Lecture/Dialogue	
2. The principles of Mathematical Theory Exploration. Sets	Q&A/Lecture/Dialogue	
3. Systematical Exploration of Lists Theory (I)	Q&A/Lecture/Dialogue	
4. Systematical Exploration of Lists Theory (II)	Q&A/Lecture/Dialogue	
5. Systematical Exploration of Binary Trees (I)	Q&A/Lecture/Dialogue	
6. Systematical Exploration of Binary Trees (II)	Q&A/Lecture/Dialogue	
7. Algorithm Synthesis Methods	Q&A/Lecture/Dialogue	
8. Algorithm Synthesis by Lazy Thinking	Q&A/Lecture/Dialogue	
9. Principles and techniques for Proof based Algorithm Synthesis	Q&A/Lecture/Dialogue	
10. Proof based Algorithm Synthesis on Lists and Binary Trees	Q&A/Lecture/Dialogue	
11. Synthesis of Sorting Algorithms on Lists and Binary Trees	Q&A/Lecture/Dialogue	
12. Algorithm Verification	Q&A/Lecture/Dialogue	
13. Revision	Q&A/Lecture/Dialogue	
14. Colloquium	Q&A/Lecture/Dialogue	
Recommended bibliography / Bibliografie		

<ol style="list-style-type: none"> 1. Isabela Drămnesc. Algorithm Synthesis and Mathematical Theory Exploration. Lecture Notes (online) https://staff.fmi.uvt.ro/~isabela.dramnesc 2. Tudor Jebelean. Automatic Theorem Proving. Lecture Notes (online) 3. Bruno Buchberger. Logic for Computer Science. Lecture Notes (online) 4. Chin-Liang Chang, Richard Char-Tung Leel. Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973. 5. Jean Gallier. Logic for Computer Science. Foundations of Automatic Theorem Proving. 6. M. Ben-Ari, Mathematical Logic for Computer Science, Springer Verlag, London, 2nd edition, 2001. 7. Z. Manna, R. Waldinger. The Logical Basis for Computer Programming, volume 1: Deductive Reasoning, Addison-Wesley, 1985. 8. I. Dramnesc, T. Jebelean. Synthesis of list algorithms by mechanical proving, Journal of Symbolic Computation, 2015, 68, 61-92. 9. I. Dramnesc, T. Jebelean, S. Stratulat. Mechanical synthesis of sorting algorithms for binary trees by logic and combinatorial techniques, Journal of Symbolic Computation, 2019, 90, 3-41. 10. I. Dramnesc, T. Jebelean. Synthesis of sorting algorithms using multisets in Theorema, Journal of Logical and Algebraic Methods in Programming, 2021, 119, 100635. 		
8.2. Seminar, lab / Seminar, laborator	Teaching/learning strategies / Metode de predare/ învățare	Remarks, details / Observații
Lab 1-7. Exemplification of the concepts introduced in the lectures.	Experiment, discussion, team work, problem based learning, homeworks, small programming projects.	Student tasks at: https://staff.fmi.uvt.ro/~isabela.dramnesc and on Google Classroom. We use Mathematica and Theorema (in online classes we will connect remotely to the computers from UVT lab rooms in order to use Mathematica and Theorema softwares).
Recommended bibliography / Bibliografie		
<ol style="list-style-type: none"> 1. Isabela Drămnesc. Algorithm Synthesis and Mathematical Theory Exploration. Lecture Notes (online) https://staff.fmi.uvt.ro/~isabela.dramnesc 2. Tudor Jebelean. Automatic Theorem Proving. Lecture Notes (online) 3. Bruno Buchberger. Logic for Computer Science. Lecture Notes (online) 4. Chin-Liang Chang, Richard Char-Tung Leel. Symbolic Logic and Mechanical Theorem Proving. Academic Press, 1973. 5. Jean Gallier. Logic for Computer Science. Foundations of Automatic Theorem Proving. 6. M. Ben-Ari, Mathematical Logic for Computer Science, Springer Verlag, London, 2nd edition, 2001. 7. Z. Manna, R. Waldinger. The Logical Basis for Computer Programming, volume 1: Deductive Reasoning, Addison-Wesley, 1985. 8. I. Dramnesc, T. Jebelean. Synthesis of list algorithms by mechanical proving, Journal of Symbolic Computation, 2015, 68, 61-92. 9. I. Dramnesc, T. Jebelean, S. Stratulat. Mechanical synthesis of sorting algorithms for binary trees by logic and combinatorial techniques, Journal of Symbolic Computation, 2019, 90, 3-41. 		

9. Correlations between the content of the course and the requirements of the IT field / Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

The content of the lecture is similar to similar standard lectures in well known and established universities.

10. Evaluation / Evaluare*

Activity / Tip de activitate	10.1. Evaluation criteria / Criterii de evaluare**	10.2. Evaluation methods / Metode de evaluare	10.3. Weight in the averaged mark / Pondere din nota finală
10.4. Lecture / Curs	10: excellent (outstanding performance with only minor errors), 8-9: very good (above the average standard but with some errors), 6-7: satisfactory (fair, but with significant shortcomings), 5: sufficient (performance meets minimum criteria), 0-4: fail (significant work has to be done).	Written exam.	60%
10.5. Seminar/ lab	Homeworks	Homework presentation during the online seminars; Small programming projects	40%
10.6. Minimal knowledge for passing / Standard minim de performanță Minimal (grade 5): general understanding of the material, ability to answer 50% of the material (as the sum of the written exam and seminar/lab tasks).			

Date/ Data completării

15.09.2021

Signature (lecture) /
Semnătura titularului de curs
Lect. Dr. Isabela Drămnesc

Signature (seminar)
Semnătura titularului de seminar
Lect. Dr. Isabela Drămnesc

Signature (director of the department)
Semnătura directorului de departament