

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		Advanced Logical and Functional Programming					
2.2. Teacher in charge of the course / Titularul activităților de curs		Conf. Dr. Mircea MARIN					
2.3. Teacher in charge of the seminar / Titularul activităților de seminar		Conf. Dr. Mircea MARIN					
2.4. Study year / Anul de studii	1	2.5. Semester / Semestrul	1	2.6. Examination type / Tipul de evaluare: E(xam)/C(olloquim)	E	2.7. Course type / Regimul disciplinei: M(andatory)/ E(lective)/ F(acultative)	M

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					30
Supplementary documentation at library or using electronic repositories / Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate					15
Preparing for laboratories, homework, reports etc. /Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					30
Exams / Examinări					10
Tutoring / Tutorat					8
3.7. Total number of hours of individual study / Total ore studiu individual	93				
3.8. Total number of hours per semester / Total ore pe semestru	135				

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

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

4.1. curriculum / de curriculum	Computational Logic, AI
4.2. skills / de competențe	C1. Programming in high level languages specific cu logical and functional programming: Prolog, Lisp, Racket, Haskell C2 Development and maintenance of computer applications. C3. Recursive thinking, top-down and bottom-up design of applications

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

5.1. for the lecture / de desfășurare a cursului	Lecture room, at least with: laptop/head projector, corresponding software
5.2. for the seminar, laboratory / de desfășurare a seminarului/laboratorului	Laboratory room, at least with: whiteboard, laptop/projector, computers with network access to internet, corresponding software.

6. Acquired skills / Competențe specifice acumulate

Professional skills / Competențe profesionale	- the ability to identify the right method to solve a complex problem, and to analyze the correctness and efficiency of the algorithm - the ability to give an algorithmic description to the identified method
Transversal skills / Competențe transversale	- the capacity to communicate knowledge concerning logical and functional programming - the capacity to apply the knowledge acquired in various other domains

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

7.1. General objective / Obiectivul general al disciplinei	- Presentation of some advanced features of functional programming, including: lexical closures, macros, abstract datatypes - Presentation of some methods o solve problems modeled by \ systems of constraints in various constraint domains: finite domains, arithmetic terms over the real or rational numbers, polynomial constraints
7.2. Specific objectives / Obiectivele specifice	The design and implementation of practical applications in some languages with well-known capabilities for functional programming, logic programming, and constraint logic programming

8. Content / Conținuturi*

8.1. Lecture / Curs	Teaching strategies / Metode de predare	Remarks, details / Observații
C1. Introductory course to Logical and Functional Programming. Programming paradigms and basic	Interactive exposure, heuristic conversation, documentation on the web, exemplification.	1 week, 2 hours All materials are available via the e-uvf.ro

notions.		facilities
C2-3. Advanced functional programming techniques in Racket.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities
C4-5. Lexical closures and abstract datatypes.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities
C6-7. Continuations. Applications.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities
C8. Logic programming: basic notions. The computational model.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	1 week, 2 hours All materials are available via the e-uvv.ro facilities
C9-10. Constraint satisfaction problems (CSP) and CSP languages.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities
C11-12. Constraint solving techniques over finite domains: finite(generate and test, standard backtracking, forward checking, lookahead, partial lookahead). Optimal solutions (branch-and-bound search techniques)	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities
C13-14. Logic programming with constraints.	Interactive exposure, problem solving, heuristic conversation, documentation on the web, exemplification.	2 weeks, 4 hours All materials are available via the e-uvv.ro facilities

Recommended bibliography / Bibliografie

1. M. Marin, V. Negru, I. Dramnesc – Principles and Practice of Functional Programming. Ed. Univ. de Vest Timisoara, 2016.
2. C. Muscalagiu - Introducere in programarea logica si limbajele de programare logica, Ed. Univ. "A.I.Cuza" Iasi, 1996
3. Bratko - PROLOG. Programming for Artificial Intelligence, Addison Wesley, third edition, 2001
4. Grillmeyer - Exploring Computer Science with Scheme, Springer, 1997.

5. P. Hudak - The Haskell School of Expression: Learning Functional Programming through Multimedia. Cambridge University Press, New York, 2000. 6. N. C. Heinze, J. Jaffar, C. Michailov, P. J. Stuckey, R. H. C. Yap - The CLP(R) Programmers Manual. 7. Dept. of Computer Science, Monash University, 1992. P. Van Hentenryck - Constraint Satisfaction in Logic Programming, M.I.T. 1989 8. T. Muller, K. Popov, C. Schulte, J. Wurtz – Constraint Programming in Oz, DFKI Oz Documentation Series, 1995. 9. D. Friedman, M. Wand, C. Hayes - Essentials of Programming Languages, second edition, MIT, 2001. 10. B. Harvey, M. Wright - Simply Scheme. Introducing Computer Science, second edition, MIT, 1999. 11. H. Abelson, G.J. Sussman, J. Sussman - Structure and Interpretation of Computer Programs, Second edition, MIT, 1996. 12. http://www.swi-prolog.org/		
8.2. Seminar, lab / Seminar, laborator	Teaching/learning strategies / Metode de predare/ învățare	Remarks, details / Observații
Applications in Prolog, Oz, CLP, Racket (former Scheme), Haskell.	Exercise, conversation and debate, modeling, project, working in organized groups.	
Recommended bibliography / Bibliografie		
1.		

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 course corresponds with curricula of other universities in Romania or the European Union. The contents of practical work (labs) meet the requirements of the local labor market.

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	Evaluation of the theoretical knowledge and practical skills	Written exam/Project/Essay	50%
	Periodic assessment of the knowledge acquired from attending the lectures	Tests, homework	20%
10.5. Seminar/ lab	Homework and project activities cover specific parts, as they were exposed during the semester, and their solution is based on laboratory activity	Individual or group project.	30%
10.6. Minimal knowledge for passing / Standard minim de performanță			
Lecture: the capacity to understand the basic principles and concepts of Logical and Functional Programming. An average level of understanding the principles of Constraint Programming.			

Lab: the ability to solve problems of average complexity

Date/ Data completării

Signature (lecture) /
Semnătura titularului de curs

Signature (seminar)
Semnătura titularului de seminar

Conf.dr. Mircea MARIN

Conf.dr. Mircea MARIN

Signature (director of the department)
Semnătura directorului de departament
Conf.dr. Victoria Iordan