



**FIȘA DISCIPLINEI**  
(licență)

**1. Date despre program**

1.1. Instituția de învățământ superior	<b>UNIVERSITATEA „VASILE ALECSANDRI” DIN BACĂU</b>
1.2. Facultatea	<b>Facultatea de Inginerie</b>
1.3. Departamentul	<b>Departamentul de Energetică și Știința Calculatoarelor</b>
1.4. Domeniul de studii	<b>Calculatoare și Tehnologia Informației</b>
1.5. Ciclul de studii	<b>Licență</b>
1.6. Programul de studii/calificarea	<b>Tehnologia Informației</b>
1.7. Forma de învățământ	<b>Învățământ cu frecvență</b>

**1. Date despre disciplină**

2.1. Denumirea disciplinei	Limbaje formale și teoria automatelor				
2.2. Titularul activităților de curs	Lect.dr. Dan POPA				
2.3. Titularul activităților de seminar	Lect.dr. Dan POPA				
2.4. Anul de studiu:	IV	2.5. Semestrul	I	2.6 Tipul de evaluare	Examen
2.7. Regimul disciplinei	Categoría formativă a disciplinei DF - fundamentală, DD - în domeniu, DS - de specialitate, DC - complementară				DS
	Categoría de opționalitate a disciplinei: DI - obligatorie (impusă), DO - opțională (la alegere), DL - facultativă (liber aleasă)				DI

**2. Timpul total estimat (ore alocate activităților didactice)**

3.1. Număr de ore pe săptămână	3	3.2. Curs	2	3.3. Seminar/Laborator/Proiect	1/0/0
3.4. Totalul de ore pe semestru din planul de învățământ	42	3.5. Curs	28	3.6. Seminar/Laborator/Proiect	14/0/0

Distribuția fondului de timp pe semestru:	ore
Studiul după manual, suport de curs, bibliografie și notițe	3
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	15
Tutoriat	0
Examinări	0
Alte activități (precizați):	0

3.7. Total ore studiu individual	33
3.8. Total ore pe semestru	75
3.9. Numărul de credite	3

**3. Precondiții (acolo unde este cazul)**

4.1. de curriculum	<ul style="list-style-type: none"> <li>Algebra și teoria multimedierilor</li> </ul>
4.2. de competențe	<ul style="list-style-type: none"> <li>Utilizare SO Linux, utilizare limbaj Haskell, utilizare limbaj C/C++</li> </ul>

**4. Condiții (acolo unde este cazul)**

5.1. de desfășurare a cursului	<ul style="list-style-type: none"> <li>Cursul se recomanda a se desfasura în sala de mari dimensiuni – fiind</li> </ul>
--------------------------------	---

	curs interfacultati - dotata cu videoproiector, tablă etc.
5.2. de desfășurare a seminarului/laboratorului/proiectului	<ul style="list-style-type: none"> <li>Seminarul se recomandă a se desfășura în sala de seminar dotata cu videoproiector, tablă și minim 30 scaune la mese.</li> </ul>

## 5. Competențe specifice acumulate

6.1. Competențe profesionale	<p>C5.1. Identificarea și descrierea instrumentelor de modelare, simulare și evaluare a performanțelor sistemelor hardware, software și de comunicații</p> <p>C5.2. Utilizarea unor cunoștințe interdisciplinare pentru asigurarea exploatarii sistemelor hardware, software și de comunicații în raport cu cerințele domeniului de aplicații</p> <p>C5.3. Utilizarea unor principii și metode de bază pentru asigurarea securității, siguranței și ușurinței în exploatarea sistemelor hardware, software și de comunicații</p> <p>C5.4. Testarea și evaluarea calitativă a caracteristicilor funcționale și nefuncționale ale sistemelor informatice, pe baza unor criterii specifice</p> <p>C5.5. Dezvoltarea de sisteme și aplicații pentru întreținerea și utilizarea de sisteme hardware, software și de comunicații</p>
6.2. Competențe transversale	<p><b>CT1</b> Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei.</p> <p><b>CT3</b> Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională.</p>

## 6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

7.1. Obiectivul general al disciplinei	<b>C4 Utilizarea bazelor teoretice ale informaticii și a modelelor formale</b> (din teoria limbajelor formale și teoria automatelor )
7.2. Obiectivele specifice	<ul style="list-style-type: none"> <li>Obiectivele specifice:</li> <li>C4.1. Definirea conceptelor și principiilor de bază ale informaticii, precum și a teoriilor și modelelor matematice (din teoria limbajelor formale și teoria automatelor ).</li> <li>C4.2. Interpretarea de modele matematice și informatice (formale).</li> <li>C4.3. Identificarea modelelor și metodelor adecvate pentru rezolvarea unor probleme reale.</li> <li>C4.5. Încorporarea de modele formale în aplicații specifice din diverse domenii.</li> </ul>

## 7. Conținuturi

Curs	Nr. ore	Metode de predare	Observații
<ul style="list-style-type: none"> <li>Introducere în gramatici și automate. Tipuri de interpretoare. Notatii algebrice folosite: mulțimi, relații, relații binare, tipuri de relații, închiderea unei relații, inducție, scheme de inducție, inducție cu mai multe reguli, recursie, recursivitatea ca instrument de realizare a interpretoarelor și compilatoarelor;</li> <li>Noțiuni de teoria grafurilor, Limbaje formale, văzute ca mulțimi de cuvinte, Reuniuni și operații cu limbaje, Gramatici ale limbajelor formale, Derivare, Exemplu: Gramatica unui limbaj de programare. Clasificarea gramaticilor, Ierarhia lui Chomsky, Echivalența gramaticilor, Limbaje decidabile.</li> </ul>	2 ore		
	2 ore		

<ul style="list-style-type: none"> <li>Automate finit(e) deterministe: Complexitatea analizei sintactice, de ce limbajele de tip 3 sunt interesante. O gramatică de tip 3. Reprezentarea grafică. Introducere informală în automate. Tranziții deterministe și tranziții nedeterministe. Automate finite deterministe și automate finite nedeterministe, Configurații și relații de tranziție, Configurație inițială, configurație finală cu stare finală, Limbaje acceptate de automate finite. Automate, reprezentări de automate, reprezentare tabelară, reprezentare graf, automate echivalente, funcționarea automatelor, mișcare, blocare, oprire și staționare. Stări accesibile, stări inaccesibile. Stări accesibile, stări inaccesibile., Algoritmul de determinare a stărilor accesibile și respectiv productive, Echivalența AFD – AFN. Transformarea AFN în AFD. Exemplu. Definiția AFD redus. Automate cu epsilon-mișcări. Transformarea unui AF pentru eliminarea lor.</li> </ul>	4 ore		
<ul style="list-style-type: none"> <li>Expresii regulate, Mulțimi regulate, Expresii regulate asociate mulțimilor, Operațiile cu automate asociate. Automate cu epsilon-mișcări care corespund expresiilor regulate. Exemple de limbaje.</li> </ul>	4 ore		
<ul style="list-style-type: none"> <li>Proprietăți algebrice ale expresiilor regulate. Construcția expresiei regulate corespunzătoare limbajului recunoscut de un automat. Metoda I: construcția inductivă a unui set de mulțimi, Metoda a II-a, Construcția unui sistem de ecuații liniare și rezolvarea acestuia.</li> </ul>	2 ore		
<ul style="list-style-type: none"> <li>Echivalența dintre limbaje regulate specificate prin gramatici și limbaje regulate recunoscute de automate. Gramatici și limbaje independente de context. Proprietăți de închidere pentru limbajele de tip 2.</li> </ul>	2 ore		
<ul style="list-style-type: none"> <li>Arbori de derivare, Analiza sintactică,</li> </ul>			
<ul style="list-style-type: none"> <li>Frontiera unui arbore de analiză sintactică, Gramatici ambigue și neambigue, Simplificarea gramaticilor dependente de context. Simboluri inaccesibile – eliminarea lor, simboluri neproductive – eliminarea lor, simboluri neutilizabile – eliminarea lor, epsilon - producții și eliminarea lor, redenumiri și eliminarea lor.</li> </ul>	4 ore		
<ul style="list-style-type: none"> <li>Recursivitate și eliminarea recursivității la stânga. Forma normală Chomsky; Greibach, Leme de pompare pentru limbaje independente de context.</li> </ul>	2 ore		
<ul style="list-style-type: none"> <li>Implementari: Implementari de parsere modulare în limbajul functional, cu clase, Haskell. Clasa Parserelor, folosirea do-notatiei. Operatii cu parsere. Construirea incrementală a parserului.</li> </ul>	2 ore		
<ul style="list-style-type: none"> <li>Implementari: Implementari de parsere în C/C++ sau alte limbaje imperative cu generatoare. Bison/Yacc împreună cu Flex/Lex. Alte generatoare.</li> </ul>	2 ore		
<ul style="list-style-type: none"> <li>Implementari: Transformarea regulilor gramaticale în funcții mutual recursive. Limitările metodei.</li> </ul>	2 ore		
Bibliografie			
1. Grigor Moldovan, <i>Limbaje Formale și Teoria automatelor</i> , Edusoft, Bacău, 2005			

2. Grune, Dick; Jacobs, J.H. Cerial, *Parsing Techniques, A Practical Guide*, 2008 Springer Verlag

3. Hudak, Paul, John Peterson, Joseph H. Fasel, *O mica introducere în Haskell 98*, traducere din limba engleză de Dan Popa, cu un cuvânt introductiv pentru ediția în limba română de Paul Hudak, București: Matrix Rom, 2011, 130 pagini, ISBN978-973-755-701-8. (Tit. orig.: *A Gentle introduction to Haskell 98* – număr de referință, din mai 1992, dedicat prezentării limbajului Haskell în revista SIGPLAN a ACM).

4. Popa, Dan V., *O introducere în Haskell 98 prin exemple*, ediția a doua republicată, Bacău: Editura Alma Mater, 2015, 312 pagini, ISBN 978-606-527-506-5.

5. Popa, Dan V., „Capitolul 10: Automate”, in *Introducere în Qt*, Bacău: Editura Alma Mater, 2015, ISBN 978-606-527-475-4

6. Popa, Dan V., „DCG-uri în Prolog (Gramatici și reguli gramaticale)”, in *Introducere în limbajul SWI-PROLOG*, Bacău: Editura Alma Mater, 2016, ISBN 978-606-527-522-5

Bibliografie minimală

O pereche formata dintr-un curs de L.F. și T.A .cum este [1] si un manual de Haskell.

Aplicații (Seminar / laborator / proiect)	Nr. ore	Metode de predare	Observații
1. Gramatica expresiilor aritmetice, implementare; exercitii.	2	Discutia frontala a problemelor.	
2. Gramatica expresiilor logice, implementare, axe descendente, recursive.	2		
3. Studiu de caz, gramatica legendelor schemelor electronice	2		
4. Introducere in QT si implementarea automatelor in C++, QT cu obiecte.	2		
5. Expresii regulate, stabilirea expresiei regulate atasate unui automat. Exerctii.	2		
6. Constructia automatelor adaptive	2		
7. Animatii cu teoria automatelor. Recapitulare.	2		

#### Bibliografie

7. Popa, Dan, *O introducere în Haskell 98 prin exemple – Editura Pim*, Iasi, oct. 2014
8. Hudak, Paul; Peterson John; Fasel Joseph, *A Gentle Introduction to Haskell 98*; online edition october, 1999
9. Hudak, Paul, John Peterson, Joseph H. Fasel, *O mica introducere în Haskell 98*, traducere din limba engleză de Dan Popa, cu un cuvânt introductiv pentru ediția în limba română de Paul Hudak, București: Matrix Rom, 2011, 130 pagini, ISBN978-973-755-701-8
10. Popa, Dan, *Practica Interpretarii Monadice* – Ed. Matrixrom, București, 2008, ISBN 978-973-755-417-8
11. [www.haskell.org/haskellwiki/Ro/Haskell](http://www.haskell.org/haskellwiki/Ro/Haskell)

#### Bibliografie minimală:

Un manual de Haskell care conține și capitolul despre parsere cum sunt [3] sau [6].

8. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

- Este coroborat fiind curs de nivel european de specialitate cu acces la o nisa de pe piata muncii: programare funcționala și software pentru citirea și analiza sintactica a datelor. Aplicatii exista și in domeniul financiar bancar. Implementarea modulelor Load/Save din meniul File al aplicatiilor face de asemenea apel al LFA.

## 9. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Evaluarea culturii generale (în domeniu) corespunde setului de întrebări nr. 1. Evaluarea cunostintelor teoretice (în domeniu) corespunde setului de întrebări nr. 2.	Examenare prin test scris.	<b>50%</b>
10.5. Seminar/laborator/proiect	Evaluarea abilitatilor de programator și rezolvitor de exercitii (în domeniu) corespunde setului de întrebări nr. 3. Evaluarea cunostintelor suplimentare nivelului 3, individual acumulate corespunde setului de întrebări nr. 4.	<b>Examinare prin întrebări care verifica existenta abilitatilor practice și experientei ce decurge din ele și (la nivelul 4) integrarea teoriei cu practica.</b>	<b>50%</b>

10.6. Standard minim de performanță:

- Nivelul 1 + Nivelul al 2-lea – a se vedea 10.4

Data completării	Semnătura titularului de curs	Semnătura titularului de seminar
<b>9 septembrie 2016</b>		

Data avizării în departament	Semnătura directorului de departament

Data aprobării în Consiliul Facultății	Semnătura decanului