



## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea “Alexandru Ioan Cuza” din Iași
1.2 Facultatea	Facultatea de Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Informatică/Licențiat în informatică

### 2. Date despre disciplină

2.1 Denumirea disciplinei	Limbaje Formale, Automate și Compilatoare						
2.2 Titularul activităților de curs	Lect. Dr. Captarencu Oana, Lect. Dr. Moruz Alex						
2.3 Titularul activităților de seminar	Lect. Dr. Captarencu Oana, Lect. Dr. Moruz Alex						
2.4 An de studiu	2	2.5 Semestru	1	2.6 Tip de evaluare	E	2.7 Regimul disciplinei	OB

\* OB – Obligatoriu / OP – Opțional

### 3. Timpul total estimat (ore pe semestru și activități didactice)

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/laborator	2
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/laborator	28
Distribuția fondului de timp					ore
Studiu după manual, suport de curs, bibliografie și altele					14
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					14
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					28
Tutoriat					
Examinări					4
Alte activități					
3.7 Total ore studiu individual					56
3.8 Total ore pe semestru					116
3.9 Număr de credite					5

### 4. Precondiții (dacă este cazul)

4.1 De curriculum	
4.2 De competențe	

### 5. Condiții (dacă este cazul)

5.1 De desfășurare a cursului	
5.2 De desfășurare a seminarului/ laboratorului	



## 6. Competențe specifice acumulate

<b>C o m p e t e n ț e p r o f e s i o n a l e</b>	<p>C1. Înțelegerea conceptelor, teoriilor și modelelor folosite în procesul de proiectare și implementare a limbajelor de programare: limbaj formal, automat finit, expresie regulată, gramatică, automat pushdown, masini Turing, forme normale pentru gramatici, analiza lexicală, analiza sintactică (algoritmi de parsare LL, LR), traducere în cod intermediar</p> <p>C2. Utilizarea modelelor și instrumentelor informatice și matematice pentru rezolvarea problemelor de limbaje formale și compilatoare</p> <p>C3. Elaborarea componentelor informatice ale unor proiecte de limbaje formale și compilatoare</p>
<b>C o m p e t e n ț e t r a n s v e r s a l e</b>	<p>CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul disciplinei, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională</p>

## 7. Obiectivele disciplinei

<b>7.1 O b i e c t i v u l g e n e r a l</b>	<ul style="list-style-type: none"><li>▪ Însușirea noțiunilor și rezultatelor fundamentale privind limbajele formale (îndeosebi a celor de tip 2 și 3), a gramaticilor, automatelor finite, expresiilor regulate și a automatelor pushdown.</li><li>▪ Cunoașterea algoritmilor fundamentali de analiză sintactică și a metodelor de traducere în cod intermediar.</li></ul>
<b>7.2 O b i e c t i v e l e s p e c i f i c e</b>	<ul style="list-style-type: none"><li>▪ Utilizarea mecanismelor de recunoaștere / generare/descriere a limbajelor de tip 2 și 3 (gramatici, automate, expresii regulate, automate pushdown)</li><li>▪ Aplicarea algoritmilor fundamentali de analiză lexicală și analiză sintactică</li><li>▪ Utilizarea instrumentelor de tip LEX/FLEX și YACC pentru generarea de analizoare sintactice</li></ul>

## 8. Conținut

<b>8.1</b>	<b>Curs</b>	<b>Metode de predare</b>	<b>Observații</b> (ore și referințe bibliografice)
1.	Limbaje formale, gramatici, ierarhia lui Chomsky; Limbaje și gramatici de tip 3.	Expunere (videoprojector)	Referințe: 3,4,5



2.	Proprietăți pentru familia de limbaje L3; Automate finite (determinate, nedeterminate, cu $\epsilon$ -tranziii), echivalența celor 3 modele;	Expunere (videoproiector)	Referințe: 3,4,5
3.	Legătura dintre gramatici de tip 3 și automate deterministe; Construcția automatului minimal.	Expunere (videoproiector)	Referințe: 3,4,5
4.	Expresii regulate; Automatul echivalent cu o expresie regulată; Gramatici și limbaje de tip 2.	Expunere (videoproiector)	Referințe: 3,4,5
5.	Arbori de derivare; Forma normală Chomsky și algoritmul CYK;	Expunere (videoproiector)	Referințe: 3,4,5
6.	Limbaje de tip 2 și automate pushdown.	Expunere (videoproiector)	Referințe: 3,4,5
7.	Analiza lexicală	Expunere (videoproiector)	Referințe: 1,2
8.	Analiza sintactică – prezentare generală	Expunere (videoproiector)	Referințe: 1,2
9.	Analiza sintactică - SLR	Expunere (videoproiector)	Referințe: 1,2
10.	Analiza sintactică - LR1	Expunere (videoproiector)	Referințe: 1,2
11.	Analiza sintactică - LALR1	Expunere (videoproiector)	Referințe: 1,2
12.	Analiza sintactică - LL1	Expunere (videoproiector)	Referințe: 1,2
13.	Traducere în cod intermediar	Expunere (online)	Referințe: 1,2
14.	Traducere în cod intermediar	Expunere (online)	Referințe: 1,2

**Bibliografie****Referințe principale:**

1. A. V. Aho, M. S. Lam, R. Sethi, J. D. Ullman: *Compilers: Principles, Techniques, and Tools*. Boston: Addison-Wesley, 2007
2. Gh. Grigoras. *Construcția compilatoarelor - Algoritmi fundamentali*, Ed. Universității Al. I. "Cuza Iasi", ISBN 973-703-084-2, 274 pg., 2005
3. Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D. (2006). *Introduction to Automata Theory, Languages, and Computation* (3rd ed.). Addison-Wesley
4. J. Toader - *Limbaje formale și automate*, Editura Matrix Rom, București, 1999.
5. J. Toader, Ștefan Andrei – *Limbaje formale și teoria automatelor. Teorie și practică*, Editura Universității "Al. I. Cuza", Iași, 2002.

**Referințe suplimentare:**

- Stoughton Alley, *Formal Language Theory*, Kansas State University, Draft of Fall 2007

8.2	Seminar / Laborator	Metode de predare	Observații (ore și referințe bibliografice)
-----	---------------------	-------------------	--



1.	Utilizarea de gramatici de tip 2 și 3 pentru generarea unor limbaje date	Rezolvare probleme, discuții	Referinte: 1
2.	Utilizarea de gramatici de tip 2 și 3 pentru generarea unor limbaje date	Rezolvare probleme, discuții	Referinte: 1
3.	Utilizarea de automate pentru recunoașterea unor limbaje date	Rezolvare probleme, discuții	Referinte: 1
4.	Algoritmul de transformare a unui automat nedeterminist/cu $\epsilon$ -tranziii în automatul determinist echivalent; Algoritmul de minimizare a unui automat determinist	Rezolvare probleme, discuții	Referinte: 1
5.	Descrierea unor limbaje formale utilizând expresii regulate; Construcția automatului echivalent cu o expresie regulată	Rezolvare probleme, discuții	Referinte: 1
6.	Aducerea de gramatici la forma normală Chomsky și utilizarea algoritmului CYK	Rezolvare probleme, discuții	Referinte: 1
7.	Utilizarea automatelor pushdown pentru recunoașterea unor limbaje de tip 2.	Rezolvare probleme, discuții	Referinte: 1
8.	Recapitulare. Evaluari	Rezolvare probleme	Referinte: 2
9.	Utilizarea aplicației Lex/Flex	Rezolvare probleme	Referinte: 2
10.	Utilizarea aplicației Yacc	Rezolvare probleme, discuții	Referinte: 3,4
11.	Prezentare proiect	discuții	Referinte: 1,2,3
12.	Rezolvare cerinte proiect	Rezolvare probleme, discuții	Referinte: 1,2,3
13.	Evaluare proiect – partea 1	Discuții (online)	Referinte: 1,5
14.	Evaluare proiect – partea 2	Discuții (online)	Referinte: 1,5

**Bibliografie**

1. Bibliografia cursului
2. Manual Flex: <http://dinosaur.compilertools.net/flex/index.html>
3. Manual Yacc: <http://dinosaur.compilertools.net/yacc/index.html>
4. Manual Bison: <http://dinosaur.compilertools.net/bison/index.htm>
5. Construcția compilatoarelor utilizând Flex și Bison: <http://foja.dcs.fmph.uniba.sk/kompilatory/docs/compiler.pdf>

**9. Coroborarea conținutului disciplinei cu așteptările reprezentanților comunității, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului**



--

**10. Evaluare**

Tip activitate	10.1 Criterii de evaluare [criteriile de punctare si clasificare, inclusiv cele de promovare]	10.2 Metode de evaluare [teste scrise, proiecte, teme, prezenta (sem/lab), activitate la tabla, bonusuri pentru activitati suplimentare, ...]	10.3 Pondere în nota finală (%)
10.4 Curs	Un test scris in sesiune. Nota minima la test trebuie sa fie 5.	Test scris (sesiune)	50%
10.5 Seminar/ Laborator	Activitatea de seminar (AS) este notată cu o notă de la 0 la 10. Activitatea la laborator (AL) este notată cu o notă de la 0 la 10. Notele obținute atât pentru AS cât și pentru AL trebuie sa fie minim 3. AS + AL trebuie sa fie minim 10.	20% din AS: activitatea din timpul seminariilor (rezolvare probleme), 80% din AS: un test scris. AL: proiect.	AS - 25% AL - 25%
<b>10.6 Standard minim de performanță [ raportate la competențele definite la punctul 7. Obiectivele disciplinei]</b>			
<ul style="list-style-type: none"><li>- Înțelegerea conceptelor, teoriilor și modelelor folosite în procesul de proiectare și implementare a limbajelor de programare</li><li>- Rezolvarea unor probleme cu grad mediu de complexitate, folosind cunoștințe de matematică și informatică (descrierea folosind mecanisme specifice a limbajelor de tip 2 si 3, aplicarea principalilor algoritmi de analiza sintactica, utilizarea uneltelor de tip Lex/Flex si Yacc)</li></ul>			

Data completării

Titular de curs

Titular de seminar

Data avizării în departament

Director de departament