

## FIȘA DISCIPLINEI

### *Limbaje formale și tehnici de compilare*

Anul universitar 2025-2026

#### 1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai
1.2. Facultatea	Matematică și Informatică
1.3. Departamentul	Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii / Calificarea	Informatică în limba germană
1.7. Forma de învățământ	Cu frecvență

#### 2. Date despre disciplină

2.1. Denumirea disciplinei	Limbaje formale și tehnici de compilare			Codul disciplinei	MLG5023		
2.2. Titularul activităților de curs	Prof. Dr. Klaus Dohmen						
2.3. Titularul activităților de seminar	Prof. Dr. Klaus Dohmen						
2.4. Anul de studiu	2	2.5. Semestrul	2	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	Obligatorie

#### 3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Număr de ore pe săptămână	6	din care: 3.2. curs	2	3.3. seminar/ laborator/proiect	2+2
3.4. Total ore din planul de învățământ	84	din care: 3.5. curs	28	3.6 seminar/laborator/proiect	56
<b>Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)</b>					<b>ore</b>
Studiul după manual, suport de curs, bibliografie și notițe (AI)					10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					10
Tutoriat (consiliere profesională)					8
Examinări					3
Alte activități					
<b>3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)</b>				<b>41</b>	
<b>3.8. Total ore pe semestru</b>				<b>150</b>	
<b>3.9. Numărul de credite</b>				<b>5</b>	

#### 4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Structuri de date și algoritmi
4.2. de competențe	Abilități de programare

#### 5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sala de curs dotată cu tablă și videoproiector
5.2. de desfășurare a seminarului/ laboratorului	Laborator dotat cu tablă și videoproiector, laptopuri

#### 6.1. Competențele specifice acumulate<sup>1</sup>

<sup>1</sup> Se poate opta pentru competențe sau pentru rezultatele învățării, respectiv pentru ambele. În cazul în care se alege o singură variantă, se va șterge tabelul aferent celeilalte opțiuni, iar opțiunea păstrată va fi numerotată cu 6.

<b>Competențe profesionale/ esențiale</b>	<p>C 4.1 Definierea conceptelor și principiilor de bază ale informaticii, precum și a teoriilor și modelelor matematice</p> <p>C 4.2 Interpretarea modelelor formale de matematică și informatică</p> <p>C 4.3 Identificarea modelelor și metodelor adecvate pentru rezolvarea problemelor reale</p> <p>C 4.4 Aplicarea simulărilor pentru investigarea comportamentului modelelor aplicate și evaluarea rezultatelor</p> <p>K4.5 Incorporarea modelelor formale în aplicații adecvate ale domeniilor specifice</p>
<b>Competențe transversale</b>	<p>TK1 Aplicarea regulilor pentru o muncă bine organizată și eficientă, pentru o atitudine responsabilă față de didactică și știință, pentru dezvoltarea creativă a propriului potențial, cu respectarea principiilor și normelor de etică profesională</p> <p>TK3 Aplicarea unor metode și tehnici eficiente de învățare, informare și cercetare, pentru dezvoltarea capacității de a pune în practică cunoștințele, de a se adapta nevoilor unei societăți dinamice și de a comunica în limba română și într-o limbă de circulație internațională</p>

## 6.2. Rezultatele învățării

<b>Cunoștințe</b>	<p>Absolventul are cunoștințe de programare, matematică, inginerie și tehnologie și are abilitățile de a le utiliza pentru a crea sisteme informatice complexe.</p> <p>Absolventul are cunoștințele necesare pentru a selecta și aplica proceduri de instruire adecvate (metode de predare) pentru a facilita procesul de dobândire a cunoștințelor.</p>
<b>Aptitudini</b>	<p>Absolventul este capabil să prezinte și să explice metode, algoritmi, paradigme și tehnici din diverse domenii ale informaticii.</p> <p>Absolventul este capabil să identifice probleme complexe și să investigheze aspectele aferente pentru a dezvolta opțiuni de soluționare și a implementa soluții.</p> <p>Absolventul va fi capabil să combine informații diverse pentru a formula soluții și a genera idei pentru dezvoltarea de noi produse și aplicații.</p>
<b>Responsabilități și autonomie</b>	<p>Absolventul are capacitatea de a aplica reguli generale la probleme specifice și de a dezvolta soluții relevante.</p> <p>Absolventul va fi capabil să selecteze și să aplice paradigme de programare (procedurale, orientate pe obiecte, funcționale) pentru a dezvolta aplicații software potrivite pentru domeniul specific al aplicației dezvoltate.</p>

## 7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

<b>7.1 Obiectivul general al disciplinei</b>	<p>Învățarea și înțelegerea modului de construire a compilatoarelor.</p> <p>Îmbunătățirea abilităților de programare.</p>
<b>7.2 Obiectivele specifice</b>	<p>Cunoștințe despre back-end-ul unui compilator.</p> <p>Însușirea conceptelor de bază ale limbajelor formale.</p> <p>Învățarea conceptelor de bază despre compilatoare.</p>

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Introducere în implementarea compilatoarelor. Gramatici și limbaje.	Prezentarea temei, discuția	
2. Analiză lexică.	Prelegere, demonstrație, discuție	
3. Gramatici regulate, automate finite.	Prelegere, demonstrație, discuție	

4. Gramatici independente de context.	Prelegere, demonstrație, discuție	
5. Automate împingătoare.	Prelegere, demonstrație, discuție	
6. Gramatici speciale – gramatici LL(k).	Prelegere, demonstrație, discuție	
7. Gramatici speciale – gramatici LR(k).	Prelegere, demonstrație, discuție	
8. Gramatici speciale – Gramatici LR(k). Analiza SLR, LR(1) și LALR.	Prelegere, demonstrație, discuție	
9. Structura compilatorului.	Prelegere, demonstrație, discuție	
10. Generator de analize lectice lex/flex.	Prelegere, demonstrație, discuție	
11. Demonstrații, construcții și aplicații. Echivalența automatelor finite cu gramaticile regulate.	Prelegere, demonstrație, discuție	
12. Demonstrații, construcții și aplicații. Proprietățile limbajelor regulate.	Prelegere, demonstrație, discuție	
13. Analiza semantică.	Prelegere, demonstrație, discuție	
14. Aplicații ale mecanismelor formale în analiza semantică.	Prelegere, demonstrație, discuție	
<p>Bibliografie în limba germană</p> <p>[1] C. WAGENKNECHT, HIELSCHER M., Formal Languages, Abstract Automata and Compilers, Vieweg Teubner, 2009.</p> <p>[2] ASTEROTH, A., BAIER, C., Informatică teoretică, o introducere în computabilitate, complexitate și limbaje formale, Pearson Studium, 2002.</p> <p>[3] HRONKOVIC, J., Informatică teoretică, limbaje formale, calculabilitate, teoria complexității, algoritmică, comunicare și criptografie, Vieweg Teubner, 2011.</p> <p>Bibliografie în limba engleză</p> <p>[1] KD COOPER, L. TORCZON - Proiectarea unui compilator, Elsevier Science &amp; Technology, 2011.</p> <p>[2] AV AHO, DJ ULLMAN - Principiile proiectării compilatoarelor, Addison-Wesley, 1978.</p>		
<b>8.2 Seminar</b>	Metode de predare	Observații
1-2. Specificarea unui limbaj de programare. Notația BNF.	Exemple, discuții	
3-4. Gramaticile, limba produsă de gramatici, gramatica unei limbi.	Exemple, discuții	
5-6. Automate finite	Exemple, discuții	
7-8. Gramatici independente de context. Analiză sintactică LL(1).	Exemple, discuții	

9-10. Analiza sintactică LR(k).	Exemple, discuții	
11-12. Proprietăți ale limbajului. Demonstrații și aplicații.	Exemple, discuții	
13-14. APD. Gramatici de tipurile 1, 2 și 3 (ierarhia Chomsky).	Exemple, discuții	

Bibliografie limba germană

- [1] C. WAGENKNECHT, HIELSCHER M., Formal Languages, Abstract Automata and Compilers, Vieweg Teubner, 2009.  
 [2] ASTEROTH, A., BAIER, C., Informatică teoretică, o introducere în computabilitate, complexitate și limbaje formale, Pearson Studium, 2002.  
 [3] HRONKOVIC, J., Informatică teoretică, limbaje formale, calculabilitate, teoria complexității, algoritmică, comunicare și criptografie, Vieweg Teubner, 2011.

Bibliografie în limba engleză

- [1] KD COOPER, L. TORCZON - Proiectarea unui compilator, Elsevier Science & Technology, 2011.  
 [2] AV AHO, DJ ULLMAN - Principiile proiectării compilatoarelor, Addison-Wesley, 1978

<b>8.3 Laborator</b>	Metoda de predare și învățare	Observații
1. Specificați un mini-limbaj de programare și implementați un analizator lexical.  Specificația limbajului de programare mini	Exemple, discuții, lucru în echipă	
2. Specificați un mini-limbaj de programare și implementați un analizator lexical.  Implementarea principalelor funcții ale analizorului.	Exemple, discuții, lucru în echipă	
3. Specificați un mini-limbaj de programare și implementați un analizator lexical.  Organizarea tabloului de simboluri.	Exemple, discuții, lucru în echipă	
4. Specificați un mini-limbaj de programare și implementați un analizator lexical.  Program principal. Testare și trimitere.	Exemple, discuții, lucru în echipă	
Automate finite: Verificarea acceptării unei secvențe de date. Alegerea structurilor de date și a arhitecturii aplicației.	Exemple, discuții, lucru în echipă	
Automate finite: implementare, testare și depunere.	Exemple, discuții, lucru în echipă	
Automate finite: Adaptarea programului pentru analiza lexicală pentru a permite utilizarea automatelor finite. Determinarea secvențelor de atomi lexicali.	Exemple, discuții, lucru în echipă	
Implementarea unui analizator sintactic: selecția structurii de date și arhitectura programului.	Exemple, discuții, lucru în echipă	

Implementarea unui analizator sintactic: principalele caracteristici.	Exemple, discuții, lucru în echipă		
Implementarea unui analizator sintactic: programul principal și integrarea modulelor.	Exemple, discuții, lucru în echipă		
Implementarea unui analizator sintactic: testare și depanare.	Exemple, discuții, lucru în echipă		
Implementarea unui analizator sintactic: Depunere.	Exemple, discuții, lucru în echipă		
Aplicarea lex/flex + yacc/bison: Implementare.	Exemple, discuții, lucru în echipă		
Aplicarea lex/flex + yacc/bison: testare și distribuire.	Exemple, discuții, lucru în echipă		
Bibliografie [1] KD COOPER, L. TORCZON - Proiectarea unui compilator, Elsevier Science & Technology, 2011. [2] Materiale online, de exemplu <a href="http://dinosaur.compilertools.net/">http://dinosaur.compilertools.net/</a>			

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

Această prelegere este ținută la universități de renume internațional în domeniul informaticii.

Conținutul cursului este considerat o parte importantă a abilităților de programare ale informaticienilor din companiile de software.

### 10. Evaluare

Tipul activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Ponderea la nota generală
10.4 Curs	Utilizarea corectă a conceptelor de bază ale prelegerii	lucrare scrisă	75%
10.5 Seminar/Laborator	Aplicarea practică a conceptelor teoretice Lucrări de laborator	Lucrare la laborator	25%
10.6 Standarde minime de performanță			
Pentru a promova examenul, trebuie obținută nota minimă 5			

## 11. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)<sup>2</sup>

*Nu se aplică.*

Data completării:  
17.04.2025

Semnătura titularului de curs

Prof. Dr. Klaus Dohmen

Semnătura titularului de seminar

Prof. Dr. Klaus Dohmen

Data avizării în departament:

...

Semnătura directorului de departament

Conf.dr. Adrian STERCA

---

<sup>2</sup> Păstrați doar etichetele care, în conformitate cu [Procedura de aplicare a etichetelor ODD în procesul academic](#), se potrivesc disciplinei și ștergeți-le pe celelalte, inclusiv eticheta generală pentru *Dezvoltare durabilă* - dacă nu se aplică. Dacă nicio etichetă nu descrie disciplina, ștergeți-le pe toate și scrieți "*Nu se aplică.*".