

FIȘA DISCIPLINEI

Aplicații ale inteligenței computaționale în ingineria software

Anul universitar 2026-2027

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Departamentul de informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclu de studii	Master
1.6. Programul de studii / Calificarea	Inginerie Software
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei	Aplicații ale inteligenței computaționale în ingineria soft	Codul disciplinei	MME8063		
2.2. Titularul activităților de curs	Prof. dr. Czibula Istvan				
2.3. Titularul activităților de seminar	Prof. dr. Czibula Istvan				
2.4. Anul de studiu	2	2.5. Semestrul	3	2.6. Tipul de evaluare	Examen
2.7. Regimul disciplinei	Opțional	2.8. Tipul disciplinei	Disciplină complementară (DC)		

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

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. seminar/ laborator/proiect	1 sem + 1 pr
3.4. Total ore din planul de învățământ	56	din care: 3.5. curs	28	3.6 seminar/laborator/proiect	28
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					26
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					36
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					35
Tutoriat (consiliere profesională)					12
Examinări					10
Alte activități					
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				119	
3.8. Total ore pe semestru				175	
3.9. Numărul de credite				7	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Inteligență Artificială
4.2. de competențe	Abilități de programare

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu videoproiector
5.2. de desfășurare a seminarului/ laboratorului	Laborator cu calculatoare; medii de dezvoltare software (.NET, Java, etc)

6.1. Competențele dobândite în urma absolvirii programului de studii (se preiau din planul de învățământ)¹

Competențe profesionale	
Codul competenței	Competență
	însușirea conceptelor matematice și informatice care să faciliteze înțelegerea, dezvoltarea, utilizarea și valorificarea metodelor și tehnicilor de inteligență computațională
	capacitatea de a aborda și rezolva probleme complexe folosind tehnici variate de inteligență computațională.
Competențe transversale	
Codul competenței	Competență
	capacitatea de analiză și sinteză a informației
	comportarea onorabilă, etică, respectarea deontologiei profesionale
	abilități de muncă în echipă, cu preluarea diferitelor roluri de execuție și conducere pentru realizarea unor proiecte

6.2. Rezultatele învățării specifice programului de studii (se preiau din planul de învățământ)¹

Rezultatele învățării vizate prin disciplină		
Codul competenței	Cunoștințe și înțelegere (Knowledge and understanding)	Abilități academice specifice (Specific academic skills)
	Studentul cunoaște norme și reguli etice și deontologice în cercetarea științifică;	Studentul este capabil să folosească limbajul de specialitate și terminologia specifică domeniului inteligenței computaționale, astfel încât să poată comunica și interacționa cu membrii unor echipe de lucru.
	Studentul cunoaște metode de modelare prin care analizează probleme din viața reală, le transpune în cerințe concrete și elaborează un model software corespunzător;	Studentul este capabil să aplice cunoștințe avansate de inteligență computațională, plecând de la studierea la un nivel ridicat de abstractizare a diferitelor sisteme, fiind capabil/ă să ofere soluții de implementare pentru aplicații la sisteme informatice complexe, integrate.
	Studentul cunoaște cerințele specifice demersului de cercetare în domeniul informaticii în general și al domeniului inteligenței computaționale în special și înțelege rolul cercetării în promovarea progresului.	Studentul are capacitatea de a lucra independent pentru obținerea informațiilor necesare proiectării, organizării, realizării și evaluării demersurilor de cercetare în domeniul inteligenței computaționale

7. Rezultatele învățării specifice disciplinei

Cunoștințe și înțelegere (Knowledge and understanding)
1. Studentul cunoaște principiile, tehnicile și aplicațiile din domeniul Ingineriei Software bazate pe căutare (SBSE)
2. Studentul înțelege stadiul curent al cercetării în domeniul SBSE

¹ Se vor prelua din Planul de învățământ al programului de studii acele competențe profesionale și/sau transversale la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa disciplinei. Pentru fiecare competență se va prelua întregul enunț, inclusiv codul competenței, cu formularea care apare în planul de învățământ, fără modificări. Dacă nu se preia nici o competență din oricare din cele două categorii, se șterge linia din tabel aferentă acelei categorii.

3. Studentul înțelege necesitatea și importanța utilizării tehnicilor inteligenței computaționale pentru rezolvarea unor probleme din Ingineria Software (SE)

Abilități academice specifice (Specific academic skills)

1. Studentul este capabil să aplice tehnici ale inteligenței computaționale pentru rezolvarea unor probleme din SE

2. Studentul este capabil să deruleze cercetări originale în SBSE

8. Conținuturi

8.1 Curs	Metode de predare - învățare	Observații ²
1. Introducere <ul style="list-style-type: none"> • Ingineria software bazată pe căutare (Search Based Software Engineering – SBSE) • Concepte de bază și probleme abordate 	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
2. Instruire automată în ingineria software <ul style="list-style-type: none"> ▪ Tehnici de instruire automată • Aplicații 	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
3. Tehnici ale inteligenței computaționale (CI) pentru înțelegerea programelor	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
4. Tehnici CI pentru refactorizare	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
5. Tehnici CI pentru detectarea și predicția defectelor software	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
6. Tehnici CI pentru testare software	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
7. Tehnici CI pentru vizualizare software	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
8. Tehnici CI pentru estimarea costului și predicția efortului de dezvoltare software	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
9. Tehnici CI pentru reutilizare software	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația 	

² De exemplu aspecte organizatorice, recomandări pentru studenți, aspecte specifice legate de curs/seminar cum ar fi invitarea unor practicieni în domeniu etc.

	<ul style="list-style-type: none"> • Demonstrația didactică 	
10. Tehnici CI pentru identificarea șabloanelor de proiectare	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația • Demonstrația didactică 	
Prezentare rapoarte de cercetare CISE	<ul style="list-style-type: none"> • Expunerea interactivă • Conversația • Evaluarea orală 	

Bibliografie

1. Czibula, I., G., Use of search techniques to software development, Editura Risoprint, ISBN 978-973-53- 0119-4, 2009 (248 pagini)
2. Mark Harman and Bryan F. Jones. Search-based software engineering. Information & Software Technology, 43(14):833-839, 2001.
3. Olaf Seng, Johannes Stammel, and David Burkhart. Search-based determination of refactorings for improving the class structure of object-oriented systems. In GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation, pages 1909-1916, New York, NY, USA, 2006. ACM Press.
4. Frank Simon, Frank Steinbruckner, and Claus Lewerentz. Metrics based refactoring. In CSMR '01: Proceedings of the Fifth European Conference on Software Maintenance and Reengineering, pages 30-38, Washington, DC, USA, 2001. IEEE Computer Society

8.2 Seminar/laborator	Metode de predare - învățare	Observații
1. Aspecte administrative.	<ul style="list-style-type: none"> • Expunerea interactivă • Explicația • Conversația 	Activitatea este structurată sub forma a 2 ore din 2 în 2 săptămâni.
2. Documentare pe Internet și Intranet în vederea alegerii temei pentru raportul de cercetare și alegerea săptămânii pentru prezentare.	<ul style="list-style-type: none"> • Documentare • Explicația • Conversația 	
<i>Un proiect software pe o team din SBSE (Proiect 1) va fi dezvoltat folosind un mediu de dezvoltare ML existent. Al doilea proiect software (Proiect 2) va fi dezvoltat și implementat. Aplicația va demonstra utilizarea tehnicilor CI pentru o sarcină specifică din ingineria software.</i>		
3. Definirea și specificarea problemei pentru Proiect 2	<ul style="list-style-type: none"> • Temă de laborator • Explicația • Conversația 	
4. Comentarii despre soluție (analiza problemei) și modelarea din perspectiva SBSE a problemei (Proiect 2). Demonstrarea Proiectului 1	<ul style="list-style-type: none"> • Temă de laborator • Explicația • Conversația 	
5. Documentația de proiectare a Proiectului 2	<ul style="list-style-type: none"> • Temă de laborator • Explicația • Conversația 	
6. Codul sursă pentru Proiectul 2. Demonstrarea proiectului 2	<ul style="list-style-type: none"> • Temă de laborator • Explicația • Conversația 	

Bibliografie

1. Czibula, I., G., Use of search techniques to software development, Editura Risoprint, ISBN 978-973-53- 0119-4, 2009 (248 pagini)
2. Mark Harman and Bryan F. Jones. Search-based software engineering. Information & Software Technology, 43(14):833-839, 2001.
3. Olaf Seng, Johannes Stammel, and David Burkhart. Search-based determination of refactorings for improving the class structure of object-oriented systems. In GECCO '06: Proceedings of the 8th annual conference on Genetic and

evolutionary computation, pages 1909-1916, New York, NY, USA, 2006. ACM Press.

4. Frank Simon, Frank Steinbruckner, and Claus Lewerentz. Metrics based refactoring. In CSMR '01: Proceedings of the Fifth European Conference on Software Maintenance and Reengineering, pages 30- 38, Washington, DC, USA, 2001. IEEE Computer Society.

9. Evaluare

Tip activitate	9.1 Criterii de evaluare ³	9.2 Metode de evaluare ⁴	9.3 Pondere din nota finală
9.4 Curs	<ul style="list-style-type: none">Va fi redactat și prezentat un raport de cercetare pe o tema din domeniul SBSE, pe baza unor rezultate recente obținute în cercetare	Evaluarea raportului de cercetare (articolul scris și prezentarea orală)	20%
	<ul style="list-style-type: none">Corectitudinea și completitudinea cunoștințelor acumulate	Examen scris (în sesiune)	40%
	<ul style="list-style-type: none">Activitate	4 absențe nemotivate sunt premise, restul se vor depuncta	10%
9.5 Seminar/Laborator	<ul style="list-style-type: none">Realizarea unui proiect software SBSE folosind un software ML opensource	Evaluare proiectului (documentare și demonstrare)	15%
	<ul style="list-style-type: none">Un proiect software din domeniul SBSE va fi implementat în întregime, fără a folosi biblioteci ML.	Evaluarea proiectului (implementare, documentare și demonstrare)	15%
10.6 Standard minim de promovare			
<ul style="list-style-type: none">Cunoașterea elementelor fundamentale de teorie. Fiecare student trebuie să demonstreze că a atins un nivel acceptabil de cunoaștere și înțelegere a domeniului SBSE, că este capabil să exprime cunoștințele într-o formă coerentă, că are capacitatea de a stabili anumite conexiuni și de a utiliza cunoștințele în rezolvarea unor probleme.Pentru promovare e necesar ca nota finală să fie minimum 5.			

11. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)⁵

Nu se aplică.

Data completării:

Semnătura titularului de curs

Semnătura titularului de seminar

07.05.2026

Prof. dr. Istvan Gergely Czibula

Prof. dr. Istvan Gergely Czibula

Data avizării în departament:

Semnătura directorului de departament

³ Criteriile de evaluare trebuie să reflecte direct rezultatele învățării vizate la nivel de program de studii, respectiv la nivel de disciplină. Mai concret, se evaluează achizițiile de învățare menționate în rezultatele anticipate ale învățării.

⁴ Se recomandă stabilirea atât a metodelor de evaluare finală, cât și a strategiei de evaluare pe parcurs.

⁵ 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".

Conf. dr. Adrian STERCA