

FIȘA DISCIPLINEI

METODE AVANSATE DE PROGRAMARE

Anul universitar 2025-2026

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	Licență
1.6. Programul de studii / Calificarea	Informatică – limba de studiu română
1.7. Forma de învățământ	Cu frecvență

2. Date despre disciplină

2.1. Denumirea disciplinei		Metode avansate de programare				Codul disciplinei	MLR5008
2.2. Titularul activităților de curs		Conf. Dr. Camelia Șerban					
2.3. Titularul activităților de seminar		Conf. Dr. Camelia Șerban					
2.4. Anul de studiu	2	2.5. Semestrul	3	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	Obligativu

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	2sem+2 lab
3.4. Total ore din planul de învățământ	84	din care: 3.5. curs	28	3.6 seminar/laborator/proiect	28sem+ 28lab
Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)					ore
Studiul după manual, suport de curs, bibliografie și notițe (AI)					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri (mai mare sau egal cu nr. total ore prevăzut în calendarul disciplinei pentru temele de control)					26
Tutoriat (consiliere profesională)					5
Examinări					5
Alte activități [de ex.: comunicare bidirecțională cu titularul de disciplină / tutorele]					-
3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)				66	
3.8. Total ore pe semestru				150	
3.9. Numărul de credite				6	

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Programare orientată obiect, Algoritmi, Structuri de date	
4.2. de competențe	Cunoștințe medii de programare într-un limbaj de programare de nivel înalt	

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu videoreteproiector	
5.2. de desfășurare a seminarului/ laboratorului	Sală de seminar cu videoproiector, Sală de laborator cu calculatoare (Medii de programare pentru Java și .NET)	

6.1. Competențele specifice acumulate¹

Competențe profesionale/esențiale	<ul style="list-style-type: none">• C1.1 Descrierea adecvată a paradigmelor de programare și a mecanismelor de limbaj specifice, precum și identificarea diferenței dintre aspectele de ordin semantic și sintactic.• C1.2 Explicarea unor aplicații soft existente, pe niveluri de abstractizare (arhitectură, pachete, clase, metode) utilizând în mod adecvat cunoștințele de bază• C1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date• C1.4 Testarea unor aplicații pe baza unor planuri de test• C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente
Competențe transversale	<ul style="list-style-type: none">• CT1 Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională• CT3 Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională

6.2. Rezultatele învățării

Cunoștințe	<p>Studentul are cunoștințe necesare pentru utilizarea calculatoarelor, dezvoltarea programelor și aplicațiilor software, procesarea informațiilor.</p> <p>Studentul are cunoștințe legate de programare, matematică, inginerie și tehnologie și are abilitățile necesare pentru a le folosi în crearea de sisteme informatice complexe.</p> <p>Studentul are cunoștințele necesare pentru proiectarea, analiza și administrarea bazelor de date.</p>
Aptitudini	<p>Studentul are abilitatea de a dezvolta, proiecta și crea noi aplicații, sisteme sau produse folosind bunele practici din domeniu.</p> <p>Studentul are aptitudinile necesare pentru conceperea programelor de calculator și analiza sistemelor software.</p> <p>Studentul are abilitatea de a aplica reguli generale unor probleme specifice și de a produce soluții relevante.</p> <p>Studentul are aptitudinile necesare pentru înțelegerea și folosirea conceptelor programării orientate obiect la dezvoltarea unor aplicații software de complexitate medie.</p> <p>Studentul are abilitatea de a înțelege și folosi șabloanele de proiectare pentru dezvoltarea aplicațiilor.</p> <p>Studentul are deprinderile necesare pentru a dezvolta aplicații cu interfețe grafice folosind șabloane arhitecturale potrivite pentru specificul aplicațiilor cu interacțiune utilizator.</p>

¹ 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.

Responsabilități și autonomie	Studentul este capabil să identifice probleme complexe și să examineze probleme conexe pentru a dezvolta opțiuni de rezolvare și implementa soluții.
	Studentul este capabil să combine informații diverse pentru a formula soluții și genera idei de dezvoltare pentru noi produse și aplicații.
	Studentul este capabil să aplice șabloane arhitecturale, șabloane de proiectare și bunele practici în domeniu pentru a proiecta aplicații software de complexitate mare.
	Studentul a dobândit capacitatea de a evalua diferite arhitecturi și soluții posibile pentru o problemă.
	Studentul are capacitatea de a alege și folosi paradigme de programare pentru realizarea de aplicații software.
	Studentul are capacitatea de a observa și obține informații din diverse surse.

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Înțelegerea și folosirea conceptelor programării orientate obiect la dezvoltarea unor aplicații soft de complexitate medie • Înțelegerea și folosirea unor șabloane de proiectare pentru dezvoltarea aplicațiilor. • Deprinderea dezvoltării unor aplicații cu interfețe grafice • Deprinderea folosirii unor frameworkuri pentru dezvoltarea aplicațiilor soft.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Să cunoască conceptele de bază ale ingineriei software • Să folosească instrumente de bază pentru construirea programelor • Să învețe limbajele Java/C# și instrumente de dezvoltare pentru programarea, executia și depanarea programelor Java/C#. • Să-și însușească principiile unei bune proiectări orientate obiect precum și a șablonelor de proiectare prezentate • Să-și însușească un stil de programare conform celor mai bune recomandări practice.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Introducere în platforma Java: platformă, sintaxa limbajului, tipuri de date primitive, tablouri, clase, interfețe, pachete, tipul enum, suprascrierea, supraîncărcarea, excepții	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
2. Colecții și tipuri generice: clase anonime, polimorfism, casting	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
3. Intrări - ieșiri (IO,NIO): stream (flux) pe octeți și pe caractere, files, canal de comunicare(channel), zonă tampon (buffer)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
4. Java 8: expresii lambda, fluxuri (streams)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație 	

	didactică	
5. GUI: Componentele Java FX, tratarea evenimentelor (curs introductiv)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
6. GUI (cont) – FXML, CSS, MVC pattern	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
7. Concurență: fire de execuție, sincronizare, taskuri asincrone - Java	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
8. Metaprogramming: reflecție, serializare -Java	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
9. XML: schema, documents - Java	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
10. Introducere în platforma .Net. Limbajul C#. Colecții în C#.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
11. LINQ	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
12. I/O. Concurență: fire de execuție, sincronizare, taskuri asincrone. (C#)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
13. GUI – C#.	<ul style="list-style-type: none"> • Expunere interactivă 	

	<ul style="list-style-type: none"> • Explicație • Conversație • Exemple • Demonstrație didactică 	
14. Recapitulare	•	
Bibliografie 1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition. 2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006 3. Eckel, B.: Thinking in Patterns with Java, 2004. MindView, Inc 4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994 5. ***, The Java Tutorial, 2013. http://download.oracle.com/javase/tutorial/		
8.2 Seminar / laborator	Metode de predare	Observații
1. Introducere în Java. Șabloanele Singleton, Factory Method, Template Method, Command, Decorator.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
2. Introducere în Java. Șabloanele Singleton, Factory Method, Template Method, Command, Decorator, Strategy.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
3. Colecții și Genericitate.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
4. I/O.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
5. Java 8.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
6. Sabloane de proiectare. Observer.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
7. GUI - FXML	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
8. Concurența. Reflectie. Responsive GUI	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
9. XML.	<ul style="list-style-type: none"> • Expunere interactivă 	

	<ul style="list-style-type: none"> • Explicație • Conversație • Exemple • Demonstrație didactică 	
10. C# Introducere. Colectii. Genericitate.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
11. LINQ	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
12. I.O/Concurența C#	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
13. Reflectie. GUI – C#.	Expunere interactivă <ul style="list-style-type: none"> • Explicație • Conversație • Exemple • Demonstrație didactică 	
14. Recapitulare	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
Bibliografie 1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition. 2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006 3. Eckel, B.: Thinking in Patterns with Java, 2004. MindView, Inc 4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994 5. ***, The Java Tutorial, 2013. http://download.oracle.com/javase/tutorial/		

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

- Cursul respecta recomandările IEEE și ACM legate de Curricula pentru specializarea Informatică.
- Cursul face parte din programul de studiu de la majoritatea universităților importante din România și din străinătate.
- Conținutul cursului este considerat de companiile soft ca fiind important pentru un nivel mediu de cunoștințe în programare.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Cunoașterea conceptelor proiectării orientate obiect și a principiilor unei bune proiectări orientate obiect. Capacitatea de a proiecta și implementa programe scrise în limbajele Java/C#	Examen scris	20%
10.5 Seminar/laborator	Capacitatea de a proiecta dezvoltă și depănă un program în limbajele Java/C#	Examen practic	40%
	Programele scrise în timpul semestrului	Documentație	40%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">• Minimum 5 la fiecare probă.• Pentru promovare este OBLIGATORIE prezenta la cel puțin 9 seminarii și 12 laboratoare. Studentii care nu au prezenta la minimum 9 seminarii și 12 laboratoare nu se pot prezenta la examen nici în sesiunea de restante.			

11. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)²

Nu se aplică.

Data completării:
15.04.2025

Semnătura titularului de curs

Conf. Dr. Camelia Șerban

Semnătura titularului de seminar

Conf. Dr. Camelia Șerban

Data avizării în departament:
...

Semnătura directorului de departament

Conf.dr. Adrian STERCA

² 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*".