

FIȘA DISCIPLINEI

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 Ciclul de studii	Licență
1.6 Programul de studiu / Calificarea	Matematică-Informatică – limba de studiu română

2. Date despre disciplină

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

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	1 sem +1lab
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
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					29
Tutoriat					5
Examinări					5
Alte activități:					-
3.7 Total ore studiu individual	69				
3.8 Total ore pe semestru	125				
3.9 Numărul de credite	5				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> Programare orientată obiect, Algoritmi, Structuri de date
4.2 de competențe	<ul style="list-style-type: none"> Cunoștințe medii de programare într-un limbaj de programare de nivel înalt

5. Condiții (acolo unde este cazul)

5.1 De desfășurare a cursului	<ul style="list-style-type: none"> Sală, plus proiector
-------------------------------	--

5.2 De desfășurare a seminarului/laboratorului	<ul style="list-style-type: none"> Sală de seminar cu videoproiector, Sală de laborator cu calculatoare (Medii de programare pentru Java si .NET)
--	--

6. Competențele specifice acumulate

Competențe profesionale	<p>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.</p> <p>C1.2 Explicarea unor aplicații soft existente, pe niveluri de abstractizare (arhitectură, pachete, clase, metode) utilizând în mod adecvat cunoștințele de bază</p> <p>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</p> <p>C1.4 Testarea unor aplicații pe baza unor planuri de test</p> <p>C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente</p>
Competențe transversale	<p>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</p> <p>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</p>

7. Obiectivele disciplinei (reieșind din grila competențelor acumulate)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> Intelegerea și folosirea conceptelor programării orientate obiect la dezvoltarea unor aplicații soft de complexitate medie Intelegerea 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	<ul style="list-style-type: none"> Expunere interactivă 	

octeți și pe caractere, files, canal de comunicare(channel), zonă tampon (buffer)	<ul style="list-style-type: none"> • Explicație • Conversație • Exemple • Demonstrație didactică 	
4. Java 8: expresii lambda, fluxuri (streams)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație 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. (Responsive GUI)	<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#.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
11. Colecții în C#. I/O	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație 	

	didactică	
12. LINQ	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
13. GUI – C#	<ul style="list-style-type: none"> • Expunere interactivă • Conversație 	
14. Recapitulare	<ul style="list-style-type: none"> • 	

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 Template Method, Factory și Command	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
2. Introducere în Java. Șabloanele Template Method, Factory și Command.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
3. Colecții și Genericitate. Sablonul Decorator	<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. Sablonul Observer. GUI	<ul style="list-style-type: none"> • Expunere interactivă • Explicație 	

	<ul style="list-style-type: none"> • Conversație • Exemple • Demonstrație didactică 	
7. GUI - FXML	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
8. Reflectie. XML	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
9. Sabloane de proiectare.	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
10. C# Introducere	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
11. Colectii. Genericitate. I.O	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
12. I.O / LINQ	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
13. GUI - C#	<p>Expunere interactivă</p> <ul style="list-style-type: none"> • Explicație • Conversație • Exemple • Demonstrație didactică 	
14. GUI - C#	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	

Bibliografie

1. Joseph Albahari and Ben Albahari, *C# 6.0 in a Nutshell*, Sixth Edition, O'Reilly, 2015.
2. Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
3. The Java Tutorial, SUN Microsystems, Inc. <http://download.oracle.com/javase/tutorial/>
4. Eckel, B.: *Thinking in Patterns with Java*, 2004. MindView, Inc
5. E. Gamma, R. Helm, R. Johnson, J. Vlissides, *Design Patterns – Elements of Reusable Object Oriented Software*, Ed. Addison Wesley, 1994

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	Corectitudinea și completitudinea cunoștințelor acumulate. Capacitatea de a proiecta și implementa programe scrise în limbajele Java/C#	Examen scris	30%
10.5 Seminar/laborator	Abilitatea de a scrie și depana un program Java	Examen practic	30%
	Programele scrise în timpul semestrului	Documentație	40%
10.6 Standard minim de performanță			
<ul style="list-style-type: none">• Minimum 5 la fiecare proba.			

Data completării

Semnătura titularului de curs

Semnătura titularului de seminar

Data avizării în departament

Semnătura directorului de departament

Prof. dr. Laura Diosan