

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai din Cluj-Napoca
1.2 Facultatea	Facultatea de Matematică și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Informatica
1.5 Ciclul de studii	Postuniversitar
1.6 Programul de studiu / Calificarea	Program postuniversitar de formare si dezvoltare profesionala in Informatica

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)	Programare orientată obiect (Java)						
2.2 Titularul activităților de curs	Lect. Dr. Camelia Șerban						
2.3 Titularul activităților de seminar	Lect. Dr. Camelia Șerban						
2.4 Anul de studiu	1	2.5 Semestrul	1	2.6. Tipul de evaluare	C	2.7 Regimul disciplinei	Obligatorie
2.8 Codul disciplinei	MLR5092						

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	2 lab
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					44
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					40
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					40
Tutoriat					12
Examinări					8
Alte activități:					0
3.7 Total ore studiu individual	144				
3.8 Total ore pe semestru	200				
3.9 Numărul de credite	7				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	<ul style="list-style-type: none"> Fundamentele Programarii si Algoritmica, Programare si 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 laborator cu calculatoare dotate cu limbajul de programare Java

6. Competențele specifice acumulate

Competențe profesionale	<p>C1.1 Descrierea adecvată a paradigmatelor 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	Să deprindă studentul cu proiectarea orientată obiect a problemelor de scară mică/mijlocie și învățarea limbajului de programare Java, precum și cu deprinderea principiilor unei bune proiectării orientate obiect
7.2 Obiectivele specifice	<ul style="list-style-type: none"> Să cunoască conceptele de baza ale programării Să cunoască conceptele de baza ale ingineriei software Să folosească instrumente de baza pentru construirea programelor Să învețe limbajul Java și instrumente de dezvoltare pentru programarea, executia și depanarea programelor Java. Să-și însușească principiile unei bune proiectări orientate obiect precum și a sabloanelor 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, enums, suprascrisura, supraîncărcarea, excepții	<ul style="list-style-type: none"> Expunere interactivă Explicație Conversație Exemple Demonstrație didactică 	
2. Collections and Generic Types: anonymous classes, polymorphism, casting	<ul style="list-style-type: none"> Expunere interactivă Explicație 	

	<ul style="list-style-type: none"> • Conversație • Exemple • Demonstrație didactică 	
3. IO,NIO: binary and character oriented streams, files, channels and buffers	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
4. GUI: Java FX components, event handling	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
5. GUI (cont)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
6. Functional programming: lambda expressions, streams	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
7. Concurrency: threads, executors, futures, exception handling, sync vs async methods, callback methods, cancellation	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
8. Metaprogramming: reflection, serialization	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
9. XML: schema, documents	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
10. Client-Server: sockets, nio sockets	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple 	

	<ul style="list-style-type: none"> • Demosntrație didactică 	
11. Security: authentication, authorization, users, roles	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
12. Database access: JDBC	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demosntrație didactică 	
12. Evaluarea proiectarii unei aplicatii soft	<ul style="list-style-type: none"> • Expunere interactivă • Conversație 	
13. Recapitulare	<ul style="list-style-type: none"> • 	
<p>Bibliografie</p> <p>1. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, Alex Buckley. The Java™ Language Specification Java SE 7 Edition.</p> <p>2. Eckel, B., Thinking in Java, 4th edition, Prentice Hall, 2006</p> <p>3. Eckel, B.: Thinking in Patterns with Java, 2004. MindView, Inc</p> <p>4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994</p> <p>5. ***, The Java Tutorial, 2013. http://download.oracle.com/javase/tutorial/</p>		
8.2 Seminar / laborator	Metode de predare	Observații
1. Java Basics, Polymorphism	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
2. Java Collections, Generics	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
3. Java IO	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
4. Java GUI	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	

5. Java GUI (cont)	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
6. Java Functional programming	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
7. Java Concurrency	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
8. Java metaprogramming	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
9. XML	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
10. Java Client-Server	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
11. Java security	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
12. Java database access	<ul style="list-style-type: none"> • Expunere interactivă • Explicație • Conversație • Exemple • Demonstrație didactică 	
13. Evaluarea proiectarii unei aplicatii soft.	<p>Expunere interactivă</p> <ul style="list-style-type: none"> • Explicație • Conversație • Exemple 	

	<ul style="list-style-type: none"> • 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

<ul style="list-style-type: none"> • 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 limbajul Java	Examen scris	40%
10.5 Seminar/laborator	Abilitatea de a scrie și depana un program Java	Examen practic	30%
	Programele scrise în timpul semestrului	Documentație	30%
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

3.05.2016

Data avizării în departament

Semnătura directorului de departament

Prof. dr. Anca Andreica