

## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Babeș-Bolyai Cluj-Napoca
1.2 Facultatea	Facultatea de Matematica și Informatică
1.3 Departamentul	Departamentul de Informatică
1.4 Domeniul de studii	Matematică
1.5 Ciclul de studii	Licență
1.6 Programul de studiu / Calificarea	Matematică Informatică - română

### 2. Date despre disciplină

2.1 Denumirea disciplinei	Metode Avansate de Programare						
2.2 Titularul activităților de curs	Conf. Dr. Ing. Florin Craciun						
2.3 Titularul activităților de seminar	Conf. Dr. Ing. Florin Craciun						
2.4 Anul de studiu	2	2.5 Semestrul	3	2.6. Tipul de evaluare	C	2.7 Regimul disciplinei	Obligativ

### 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	1sem+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					20
Tutoriat					5
Examinări					14
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"> <li>Programare orientată obiect, Algoritmi, Structuri de date</li> </ul>
4.2 de competențe	<ul style="list-style-type: none"> <li>Abilități de programare</li> </ul>

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

5.1 De desfășurare a cursului	<ul style="list-style-type: none"> <li>proiector</li> </ul>
5.2 De desfășurare a	<ul style="list-style-type: none"> <li>Laboratoare cu computere având instalate mediile Java și C#</li> </ul>

**6. Competențele specifice acumulate**

<b>Competențe profesionale</b>	<ul style="list-style-type: none"> <li>• Cunoștințe, înțelegerea și utilizarea conceptelor analizei și proiectării orientate obiect</li> <li>• Abilitatea de a lucra independent și/sau în echipă pentru a rezolva problemele</li> <li>• Abilități bune de programare în limbajele orientate obiect cu precădere Java și C#</li> </ul>
<b>Competențe transversale</b>	<ul style="list-style-type: none"> <li>• Abilitatea de a aplica șabloanele de proiectare în diferite contexte</li> <li>• Abilitatea de a construi proiecte software urmind principalii pași din dezvoltarea aplicațiilor software.</li> <li>• Abilitatea de a crea proiecte cu o delimitare bine definită a nivelelor arhitecturale</li> </ul>

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

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> <li>• Fiecare student va trebui să acumuleze un nivel rezonabil de cunoștințe din topica cursului, ca este capabil să utilizeze aceste cunoștințe într-o manieră coerentă, ca a dobândit abilități corecte de analiză, proiectare și implementare bazate pe șabloanele de proiectare și paradigma orientată obiect</li> </ul>
7.2 Obiectivele specifice	<ul style="list-style-type: none"> <li>• Studenții trebuie să fie capabili să utilizeze limbajele Java și C#, șabloanele de proiectare și conceptele orientate obiect pentru analiză și proiectarea programelor.</li> </ul>

**8. Conținuturi**

8.1 Curs	Metode de predare	Observații
1. - Introducere în platforma Java și platforma .Net. Concepte orientate obiect de bază. Elemente de bază în Java și C#: expresii, structuri, tipuri primitive de date	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
2. Fundamente orientate obiect în programarea în Java și C#: Clase și obiecte, Tablouri, enumerări, mostenire, interfețe, clase abstracte, clase imbricate	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
3. Fundamente orientate obiect în programarea în Java și C#: tipuri referință, tipuri valoare, polimorfism, suprascrierea, supraincarcarea, casting	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
4. Fundamente orientate obiect în programarea în Java și C#: încapsulare, modificatori de acces, pachete Java, namespace în C#, proprietăți C#, indexi C#, delegare și evenimente în C#	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
5. Tratarea excepțiilor în Java și C#	Expunerea,	

	conversația, dezbateră, problematizarea, descoperirea	
6. Cod reutilizabil utilizând Reusable Generics Types in Java și C#: - Java Generics: Generics Types, Raw Types, Generic Methods, Bounded Type Parameters, Wildcards: Upper Bounded, Unbounded, Lower Bounded, Subtyping - C# Generics: Generic Types, Generic Constraints, Subtyping, Generic Methods	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
7. Biblioteci I/O in Java și C#	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
8. Reflection și aplicațiile aferente in Java și C# - serializare	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
9. Colectii in Java și C#: clase, interfețe, algoritmi	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
10. Concurrent Programming in Java și C# - multithreading - Threads - Synchronization	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
11. Concurrent Programming in Java și C# - Thread interference - Memory consistency errors - Synchronization: locks, synchronized methods, synchronized statements - Atomic access - Immutable objects - Liveness: Deadlock, Starvation, Livelock	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
12. Concurrent Programming in Java și C# - High-Level Concurrency - Mutex - Semaphore - Signaling - Synchronization Contexts	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
13. Interfețe GUI in Java și C# - Event-driven programming - Java Swing	Expunerea, conversația, dezbateră, problematizarea, descoperirea	
14. Interfețe GUI in Java și C# - Java Swing - C# Windows Forms	Expunerea, conversația, dezbateră, problematizarea,	

## 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. Joseph Albahari and Ben Albahari, C# 4.0 in a Nutshell, Fourth Edition, O’Reilly, 2010
6. \*\*\*, Microsoft Developer Network, Microsoft Inc., <http://msdn.microsoft.com/>
7. \*\*\*, The Java Tutorial, 2013. <http://download.oracle.com/javase/tutorial/>

**8.2 Seminar****Metode de predare****Observații**

- |   |  |   |
|---|--|---|
| 1. Model-View-Controller pattern, ADT implementat cu Arrays si Linked Lists in Java | Dialogul, dezbateră, studiul de caz, exemple, demonstrații | Exista cate un seminar la fiecare doua saptamani, |
| 2. Model-View-Controller pattern, ADT implementat cu Arrays si Linked Lists in C#   | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |
| 3. Polymorphism, Exception Handling in Java si C#                                   | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |
| 4. Generic Types in Java si C#  | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |
| 5. Reflection in Java si C#, Proxy pattern si Factory pattern in C# si Java         | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |
| 6. Observer pattern in Java si C#, Event-driven programming                         | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |
| 7. Interfete GUI in Java si C#  | Dialogul, dezbateră, studiul de caz, exemple, demonstrații |   |

## Bibliography

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. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994

4. Joseph Albahari and Ben Albahari, <i>C# 4.0 in a Nutshell</i> , Fourth Edition, O'Reilley, 2010		
5. ***, Microsoft Developer Network, Microsoft Inc., <a href="http://msdn.microsoft.com/">http://msdn.microsoft.com/</a>		
6. ***, The Java Tutorial, 2013. <a href="http://download.oracle.com/javase/tutorial/">http://download.oracle.com/javase/tutorial/</a>		
Laborator	Metode de predare	Observatii
1. Alocare proiect. Programe simple in Java si C#. Java Eclipse and Visual Studio for C#(2 saptamini)		Studentii lucreaza pe acelasi proiect implementat in Java si in C# tot semestru. Fiecare laborator se refera la acelasi proiect. Fiecare tema este evaluata separat. Pt fiecare saptamina de intirziere la predarea temei se scad 0.5 puncte.
2. Model-View-Controller pattern pentru organizarea proiectului. Implementare ADT in Java si C# (2 saptamini)		
3. Tratarea erorilor utilizind exceptii in Java si C# (1 saptamina)		
4. Generalizare cod utilizind generics types in Java si C# (2 saptamini)		
5. Test de verificare		
6. Operatii I/O si serializare in Java si C# (2 saptamini)		
7. Utilizare de colectii in Java si C# (1 saptamina)		
8. Observer pattern in Java si C# (2 saptamini)		
9. GUI in Java and C# (2 saptamini)		
Bibliografie		
1. ***, Microsoft Developer Network, Microsoft Inc., <a href="http://msdn.microsoft.com/">http://msdn.microsoft.com/</a>		
2. ***, The Java Tutorial, 2013. <a href="http://download.oracle.com/javase/tutorial/">http://download.oracle.com/javase/tutorial/</a>		

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

<ul style="list-style-type: none"> <li>Acest curs exista in programul de studiu al tuturor universitatilor importante din Romania si strainatate</li> <li>Acest curs asigura cunostintele de baza pe care orice programator trebuie sa la aiba</li> </ul>
---

### 10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- cunoasterea principiilor de baza ale domeniului	Examen scris	20.00%
		Examen practic	30.00%
10.5 Seminar/laborator	- aplicarea acestor concepte in rezolvarea de	Teme laborator	35.00%
		Test practic	15.00%

	probleme		
10.6 Standard minim de performanță			
<ul style="list-style-type: none"> <li>Pentru promovare este necesara obtinerea notei minim 5 la examenul scris, examenul practic si temele de laborator</li> </ul>			

Data completării

.....

Semnătura titularului de curs

Conf.Dr.Ing. Florin Craciun

Semnătura titularului de seminar

Conf.Dr.Ing. Florin Craciun

Data avizării în departament

.....

Semnătura directorului de departament

.....