

## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Automatică și Calculatoare
1.3 Departamentul	Calculatoare
1.4 Domeniul de studii	Calculatoare și Tehnologia Informației
1.5 Ciclul de studii	Master
1.6 Programul de studii / Calificarea	Complemente de Știința Calculatoarelor / Master
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	9.17

### 2. Date despre disciplină

2.1 Denumirea disciplinei	<b>Tehnici de Programare Fundamentale</b>				
2.2 Titularii de curs	Conf. dr. ing. Cristina Pop - <a href="mailto:Cristina.Pop@cs.utcluj.ro">Cristina.Pop@cs.utcluj.ro</a>				
2.3 Titularul / Titularii activităților de seminar / laborator / proiect	Conf. dr. ing. Cristina Pop - <a href="mailto:Cristina.Pop@cs.utcluj.ro">Cristina.Pop@cs.utcluj.ro</a>				
2.4 Anul de studiu	I	2.5 Semestrul	2	2.6 Tipul de evaluare ( E – examen, C – colocviu, V – verificare)	E
2.7 Regimul disciplinei	DA – de aprofundare, DS – de sinteza, DC – complementară				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DOp

### 3. Timpul total estimat

3.1 Număr de ore pe săptămână	4	din care:	Curs	2	Seminar		Laborator	2	Proiect	
3.2 Număr de ore pe semestru	56	din care:	Curs	28	Seminar		Laborator	28	Proiect	
3.3 Distribuția fondului de timp (ore pe semestru) pentru:										
(a) Studiul după manual, suport de curs, bibliografie și notițe										10
(b) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren										10
(c) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri										20
(d) Tutoriat										
(e) Examinări										4
(f) Alte activități:										0
3.4 Total ore studiu individual (suma (3.3(a)...3.3(f)))							44			
3.5 Total ore pe semestru (3.2+3.4)							100			
3.6 Numărul de credite							4			

### 4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Programarea calculatoarelor, Baze de date
4.2 de competențe	Cunoștințe legate de programare orientată pe obiect

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

5.1. de desfășurare a cursului	Tabla, proiector, calculator; site-ul web pentru materiale de curs
5.2. de desfășurare a seminarului / laboratorului / proiectului	Tabla, proiector, calculator, internet, software specific; site-ul web pentru materiale de laborator

### 6. Competențele specifice acumulate

6.1 Competențe profesionale	<b>C4 - Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații</b> <ul style="list-style-type: none"> <li>• <b>C4.1</b> - Identificarea și descrierea elementelor definitorii ale performanțelor sistemelor hardware, software și de comunicații</li> <li>• <b>C4.2</b> - Explicarea interacțiunii factorilor care determină performanțele sistemelor hardware, software și de comunicații</li> <li>• <b>C4.3</b> - Aplicarea metodelor și principiilor de bază pentru creșterea performanțelor sistemelor hardware, software și de comunicații</li> <li>• <b>C4.4</b> - Alegerea criteriilor și metodelor de evaluare a performanțelor sistemelor hardware, software și de comunicații</li> <li>• <b>C4.5</b> - Dezvoltarea de soluții profesionale pentru sisteme hardware, software și de comunicații bazate pe creșterea performanțelor</li> </ul>
6.2 Competențe transversale	N/A

## 7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	Cunosterea si aplicarea tehnicilor de programare orientate pe obiect in dezvoltarea aplicatiilor software
7.2 Obiectivele specifice	<ul style="list-style-type: none"> <li>- Sa poata utiliza tehnici de programare pentru proiectarea claselor si interfetelor, incluzind contracte si invarianti</li> <li>- Sa poata utiliza tehnici de programare pentru reutilizarea codului folosind mostenirea si polimorfismul</li> <li>- Sa poata utiliza tehnici de programare bazate pe generics si stream-uri pentru procesarea colectiilor</li> <li>- Sa poata utiliza tehnici de programare pentru reflection, design pattern-uri si framework-uri in vederea reutilizarii solutiilor de proiectare</li> <li>- Sa aplice principiile de proiectare SOLID si thread-urile in Java</li> <li>- Sa poata utiliza tehnici de programare pentru reflectie si bazate pe evenimente</li> <li>- Sa poata utiliza tehnici de programare orientate-obiect si functionale intr-o abordare integrata pentru dezvoltarea de programe flexibile si eficiente</li> <li>- Sa poata utiliza expresiile lambda si sa poata efectua operatii de procesare a stream-urilor</li> </ul>

## 8. Conținuturi

8.1 Curs	Nr.ore	Metode de predare	Observații
Introducere - Procesul de dezvoltare software si paradigme de programare	2	- Folosirea metodelor multimedia de predare si acces la Internet - Prezentarea cursurilor si discutii fata in fata si/sau online folosind platforma Microsoft Teams si site-ul web al cursului. - Intrebari adresate studentilor in timpul cursului.	
Proiectare - Diagrame UML	2		
Paradigme de programare orientate pe obiect	2		
Tehnici de programare cu thread-uri	2		
Tehnici de programare cu clase abstracte si interfete	2		
Tehnici de compunere si reflectie	2		
Tehnici de design al claselor	2		
Tehnici de programare folosind contracte si invarianti	2		
Principii SOLID, Inversion of Control si framework-uri	2		
Flexibilitate si reutilizare folosind design pattern-uri	2		
Tehnici de programare generice	2		
Expresii lambda si procesarea stream-urilor	2		

<b>Bibliografie</b> 1. B. Eckel, On Java 8, MindView LLC, 2017 2. E. Gamma, R. Helm, R. Johnson, J. Vlissides - Design Patterns, Addison Wesley Professional, 1994 3. K. Sharan, P. Späth, More Java 17: An In-Depth Exploration of the Java Language and Its Features 3rd Edition, Apress, 2021 4. R. Urma, M. Fusco, A. Mycroft, Modern Java in Action: Lambdas, streams, functional and reactive programming, 2nd Edition, Manning, 2018 5. Note de curs online puse la dispozitie de titularii de curs 6. Online : - <a href="http://docs.oracle.com/javase/tutorial/index.html">http://docs.oracle.com/javase/tutorial/index.html</a> - <a href="http://stackoverflow.com/">http://stackoverflow.com/</a>			
<b>8.2 Aplicații (seminar/laborator/proiect)*</b>	Nr.ore	Metode de predare	Observații
Introducere – resurse si cerinte pentru laborator	2	Scurta prezentare a temelor de laborator, discutii pe baza temelor, implementarea temelor pe calculator, discutii si evaluari	N/A
Tema 1 – Tehnici de Programare cu mostenire si polimorfism	4		
Tema 2 - Tehnici de Programare cu fire de executie	4		
Tema 3 - Tehnici de Programare cu baze de date, design pattern-uri si tehnici de reflectie	6		
Tema 4 - Tehnici de Programare cu Java Collection Framework, expresii lambda si procesare de stream-uri	6		
Evaluare Laborator	4		
Bibliografie ( <i>bibliografia minimală pentru aplicații conținând cel puțin o lucrare bibliografică de referință a disciplinei care există la dispoziția studenților într-un număr de exemplare corespunzător</i> ) 1. Online : - <a href="http://docs.oracle.com/javase/tutorial/index.html">http://docs.oracle.com/javase/tutorial/index.html</a> - <a href="http://stackoverflow.com/">http://stackoverflow.com/</a>			

\*Se vor preciza, după caz: tematica seminarilor, lucrările de laborator, tematica și etapele proiectului.

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

Este o disciplină a domeniului “Calculatoare și Tehnologia Informației”. Ea îi instruește pe studenți in aplicarea tehnicilor de programare orientate pe obiect in proiectarea si implementarea aplicatiilor software. Conținutul disciplinei a fost stabilit pe baza analizei disciplinelor echivalente de la alte universități precum si pe baza cerintelor angajatorilor IT din România. De asemenea continutul disciplinei a fost evaluat de agenții guvernamentale românești (CNEAA și ARACIS).

### 10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Cunoasterea si utilizarea tehnicilor de programare prezentate in cadrul cursurilor; prezenta si interactiunea in cadrul cursurilor.	Examen scris	50%
Seminar	-	-	-
Laborator	- Abilitatea de proiectare si implementare a programelor orientate pe obiect. - Abilitatea de a aplica tehnici de programare in practica. - Calitatea codului scris si a documentației temelor. - Activitatea si prezenta in cadrul sesiunilor de laborator.	Evaluarea temelor de laborator pe parcursul semestrului	50%
Proiect	-	-	-

Standard minim de performanță:

Sa poata utiliza tehnici de programare orientate pe obiect in proiectarea si implementarea aplicatiilor software  
Calcul nota disciplina: 50% laborator + 50% examen final

Conditii de participare la examenul final: Laborator  $\geq$  5

Predarea la timp a tuturor lucrarilor de laborator si minim nota 5 pe fiecare lucrare; prezenta la cel putin 11 lucrari de laborator.

Conditii de promovare: Examen final  $\geq$  5

Conditii de predare teme de laborator restante: intr-o sesiune de restante un student poate preda 1 tema de laborator dintre cele nefinalizate in timpul semestrului.

<b>Data completării:</b>	<b>Titulari</b>	<b>Titlu Prenume NUME</b>	<b>Semnătura</b>
04.06.2024	Curs	Conf.dr.ing. Cristina-Bianca Pop	
	Aplicații	Conf.dr.ing. Cristina-Bianca Pop	

Data avizării în Consiliul Departamentului Calculatoare 20.02.2024	Director Departament, Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare 22.02.2024	Decan, Prof.dr.ing. Mihaela Dînșoreanu

