

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	Licență
1.6 Programul de studii / Calificarea	Calculatoare și Tehnologia Informației/ Inginer
1.7 Forma de învățământ	IF – învățământ cu frecvență
1.8 Codul disciplinei	25.

2. Date despre disciplină

2.1 Denumirea disciplinei	Tehnici de Programare Fundamentale				
2.2 Titularii de curs	Prof. dr. ing. Tudor Cioară – Tudor.Cioara@cs.utcluj.ro S.I.dr.ing. Cristina.Pop – Cristina.Pop@cs.utcluj.ro				
2.3 Titularul/Titularii activităților de seminar/laborator/proiect	Prof. dr. ing. Tudor Cioară – Tudor.Cioara@cs.utcluj.ro S.I.dr.ing. Cristina.Pop – Cristina.Pop@cs.utcluj.ro S.I.dr.ing. Marcel Antal – Marcel.Antal@cs.utcluj.ro S.I.dr.ing. Claudia Daniela Antal – Claudia.Pop@cs.utcluj.ro As.drd.ing. Dorin Moldovan – Dorin.Moldovan@cs.utcluj.ro				
2.4 Anul de studiu	2	2.5 Semestrul	4	2.6 Tipul de evaluare (E – examen, C – colocviu, V – verificare)	E
2.7 Regimul disciplinei	DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară				DF
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

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:										
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	Bazele Programării Orientate pe Obiect, Structuri de Date și Algoritmi
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; <i>din 16.03.2020 se folosește mediul de predare online, platforma Skype for Business, site-ul web pentru materiale de curs: http://coned.utcluj.ro/~salomie/PT_Lic</i>
5.2. de desfășurare a laboratorului	Calculatoare, software specific; <i>din 16.03.2020 se folosește mediul online, platformele Skype for Business și Discord precum și site-ul web pentru materiale de laborator: http://coned.utcluj.ro/~salomie/PT_Lic</i>

6. Competențele specifice acumulate

6.1 Competențe profesionale	C4 - Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații C4.1 - Identificarea și descrierea elementelor definitorii ale performanțelor
-----------------------------	--

	<p>sistemelor hardware, software și de comunicații</p> <p>C4.2 - Explicarea interacțiunii factorilor care determină performanțele sistemelor hardware, software și de comunicații</p> <p>C4.3 - Aplicarea metodelor și principiilor de bază pentru creșterea performanțelor sistemelor hardware, software și de comunicații</p> <p>C4.4 - Alegerea criteriilor și metodelor de evaluare a performanțelor sistemelor hardware, software și de comunicații</p> <p>C4.5 - Dezvoltarea de soluții profesionale pentru sisteme hardware, software și de comunicații bazate pe creșterea performanțelor</p>
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"> - Sa poata utiliza tehnici de programare pentru proiectarea claselor si interfetelor, incluzind contracte si invariant - Sa poata utiliza tehnici de programare pentru reutilizarea codului folosind mostenire si polimorfism - Sa poata utiliza tehnici de programare generice pentru procesarea colectiilor - Sa poata utiliza tehnici de programare pentru reflectie si bazate pe evenimente - Sa poata utiliza tehnici de programare orientate-obiect si functionale intr-o abordare integrata pentru dezvoltarea de programe flexibile si eficiente - Sa poata utiliza sabloane de proiectare si framework-uri pentru reutilizarea solutiilor de proiectare - Sa poata utiliza tehnici de programare pentru performanta si intretinere software

8. Conținuturi

8.1 Curs	Nr.ore	Metode de predare	Observații
Tehnici de proiectare si implementare pentru clase	2	Folosirea metodelor multimedia de predare si acces la Internet; prezentari online folosind platforma Skype for Business, site-ul web al cursului, discutii Studentii sunt invitati sa colaboreze la proiectele de cercetare Ore de consultatii in timpul semestrului si inaintea examenului	N/A
Tehnici de programare cu clase si interfete	2		
Tehnici de programare folosind mostenirea si polimorfismul	2		
Tehnici de programare folosind contracte si invariant	2		
Tehnici de programare generice	2		
Tehnici de programare reflexive	2		
Tehnici de programare ghidate de evenimente	2		
Tehnici de programare a colectiilor	2		
Tehnici de programare pentru concurenta si multithreading	2		
Flexibilitate si reutilizare prin design patterns	2		
Flexibilitate si reutilizare prin frameworks	2		
Expresii lambda si procesarea stream-urilor	2		
Tehnici de programare multiparadigma (OO si functional)	2		
Tehnici de programare pentru eficienta si performanta	2		
Bibliografie (<i>bibliografia minimală a disciplinei conținând cel puțin o lucrare bibliografică de referință a disciplinei, care există la dispoziția studenților într-un număr de exemplare corespunzător</i>)			
1 Ioan Salomie - Tehnici Orientate Obiect, Editura Albastra, Microinformatica, 1995			
2. Eric Gamma, Helm, Johnson, Vlissides - Design Patterns, Addison Wesley, 1995 (tradus in Romana de Editura Teora ca "Sabloane de Proiectare")			
3. Joshua Bloch - Effective Java, 2/e Addison Wesley, 2008			
4. Steve McConnell – Code Complete, 2/e, Microsoft Press, 2004			
5. Ioan Salomie, Note de Curs, http://www.coned.utcluj.ro/~salomie/TP			
6. Online :			
- http://docs.oracle.com/javase/tutorial/index.html			
- http://stackoverflow.com/			
8.2 Aplicații (seminar/laborator/proiect)*	Nr.ore	Metode de predare	Observații
Introducere – resurse si cerinte pentru laborator	2	Scurta prezentare a	-N/A

Tema 1 – Tehnici de Programare cu mostenite si polimorfism	4	temelor de laborator, discutii pe baza temelor, implementarea temelor pe calculator, discutii si evaluari in mediile online Discord, GitLab si Skype for Business
Tema 2 - Tehnici de Programare cu contracte si invarianti	4	
Tema 3 - Tehnici de Programare cu fire de executie	4	
Tema 4 - Tehnici de Programare cu sabloane de proiectare	4	
Tema 5 - Tehnici de Programare cu generics si Java Collection Framework	4	
Tema 6 – Tehnici de programare multiparadigma	4	
Evaluare Laborator	2	
Bibliografie (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)		
1. Online :		
- http://docs.oracle.com/javase/tutorial/index.html		
- http://stackoverflow.com/		

* Se vor preciza, după caz: tematica seminariilor, 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 în aplicarea tehnicilor de programare orientate pe obiect în proiectarea și implementarea aplicațiilor software. Conținutul disciplinei a fost stabilit pe baza analizei disciplinelor echivalente de la alte universități precum și pe baza cerințelor angajatorilor IT din România. De asemenea conținutul 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	- Abilitati de utilizare a tehnicilor de programare orientate pe obiect in proiectarea si implementarea aplicatiilor software - Abilitati de utilizare a sabloanelor de proiectare si a framework-urilor pentru reutilizarea solutiilor de proiectare - Abilitati de rezolvare a problemelor utilizand tehnici de programare orientate pe obiect	Examen scris supervizat online folosind platforma Skype for Business	60%
Seminar			
Laborator	- Abilități de rezolvare a problemelor utilizand tehnici de programare orientate pe obiect - Prezență, Activitate	Evaluare teme pe parcursul semestrului folosind platformele Discord si GitLab, precum si evaluare in examenul scris online supervizat prin platforma Skype for Business	40%
Proiect			

Standard minim de performanță:

- Sa poata utiliza tehnici de programare orientate pe obiect in proiectarea si implementarea aplicatiilor software

Calcul nota disciplina: 40% laborator + 60% examen final

Conditii de participare la examenul final: Laborator ≥ 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 ≥ 5

-minim nota 5 la examen.

Data completării:	Titulari	Titlu Prenume NUME	Semnătura
	Curs	Prof. Dr. Ing. Tudor Cioara	
	Aplicații	S.l.dr.ing. Cristina Pop	
		Prof. dr. ing. Tudor Cioară	
		S.l.dr.ing. Cristina Pop	
		S.l.dr.ing. Marcel Antal	
		S.l. dr.ing. Claudia Antal	
		As.Drd.ing Dorin Moldovan	

Data avizării în Consiliul Departamentului Calculatoare	Director Departament Prof.dr.ing. Rodica Potolea
Data aprobării în Consiliul Facultății de Automatică și Calculatoare	Decan Prof.dr.ing. Liviu Miclea