

## FIȘA DISCIPLINEI

(licență)

### 1. Date despre program

Instituția de învățământ superior	Universitatea „Ștefan cel Mare” din Suceava
Facultatea	Inginerie Electrică și Știința Calculatoarelor
Departamentul	Calculatoare
Domeniul de studii	Calculatoare și tehnologia informației
Ciclul de studii	Licență
Programul de studii	Program de conversie profesională – Tehnologia informației și comunicațiilor / Profesor TIC

### 2. Date despre disciplină

Denumirea disciplinei	PARADIGME DE PROGRAMARE				
Titularul activităților de curs	as.univ.dr.ing. Alexandru-Ionuț ȘIEAN				
Titularul activităților aplicative	as.univ.dr.ing. Alexandru-Ionuț ȘIEAN				
Anul de studiu	II	Semestrul	3	Tipul de evaluare	Examen
Regimul disciplinei	Categorია formativă a disciplinei DF - fundamentală, DD - în domeniu, DS - de specialitate, DC – complementară				DS
	Categorია de opționalitate a disciplinei: DI - impusă, DO - opțională, DF - facultativă				DO

### 3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	4	Curs	2	Seminar	-	Laborator/lucrări practice	2	Proiect	-
I b) Totalul de ore pe semestru din planul de învățământ	56	Curs	28	Seminar	-	Laborator/lucrări practice	28	Proiect	-

II Distribuția fondului de timp pe semestru:	ore
II a) Studiul după manual, suport de curs, bibliografie și notițe	39
II b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	39
II c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	39
II d) Tutoriat	
III Examinări	2
IV Alte activități (precizați):	

Total ore studiu individual II (a+b+c+d)	117
Total ore pe semestru (Ib+II+III+IV)	175
Numărul de credite	7

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

Curriculum	• Nu este cazul
Competențe	• Nu este cazul

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

Desfășurare a cursului	• PC, videoproiector • Suporturi electronice pentru unitatea de curs, prezentări, aplicații demonstrative	
Desfășurare aplicații	Seminar	•
	Laborator/lucrări practice	• PC, videoproiector • Suporturi electronice pentru laborator, prezentări, aplicații demonstrative • Software Visual Studio Code.
	Proiect	•

### 6. Competențe specifice acumulate

Competențe	• C1. Operarea cu fundamente științifice și ale informaticii,
------------	---

profesionale	<ul style="list-style-type: none"> <li>• C2. Proiectarea componentelor software,</li> <li>• C3. Soluționarea problemelor foloosind instrumentele tehnologiei informației și comunicațiilor,</li> <li>• C4. Îmbunătățirea performanțelor sistemelor informatice.</li> </ul>
Competențe transversale	<ul style="list-style-type: none"> <li>• CT1. Comportarea onorabilă, respectabilă, etică, în spiritul legii pentru a asigura reputația profesiei,</li> <li>• CT3. Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale și de cultură organizațională.</li> </ul>

#### 7. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

Obiectivul general al disciplinei: <i>Cunoșterea și înțelegerea rolului calculatoarelor în societatea actuală</i>	<ul style="list-style-type: none"> <li>• Înțelegerea fundamentelor matematice ale TIC.</li> <li>• Oferirea unei perspective de ansamblu asupra TIC.</li> </ul>
--	--

#### 8. Conținuturi

Curs	Nr. ore	Metode de predare	Observații
<b>1: Introducere în Paradigmele de Programare</b> <ul style="list-style-type: none"> <li>• Ce este o paradigmă de programare?</li> <li>• Principalele paradigme (imperativă, orientată pe obiecte, funcțională, logică).</li> <li>• Exemple de limbaje de programare.</li> </ul>	2	expunerea, prelegerea, dezbateri, problematizarea, conversația, demonstrația	
<b>2: Programarea Imperativă</b> <ul style="list-style-type: none"> <li>• Ce este programarea imperativă?</li> <li>• Structuri de control: bucle, condiționale.</li> <li>• Exemple simple în Python.</li> </ul>	2		
<b>3: Programarea Procedurală</b> <ul style="list-style-type: none"> <li>• Organizarea codului în funcții.</li> <li>• Parametri și valori returnate.</li> <li>• Aplicații simple cu funcții în Python.</li> </ul>	2		
<b>4: Programarea Orientată pe Obiecte – Partea 1</b> <ul style="list-style-type: none"> <li>• Noțiuni de bază: clase și obiecte.</li> <li>• Exemple simple de clase și obiecte în Python.</li> </ul>	2		
<b>5: Programarea Orientată pe Obiecte – Partea 2</b> <ul style="list-style-type: none"> <li>• Moștenirea și polimorfismul.</li> <li>• Exemple practice.</li> </ul>	2		
<b>6: Introducere în Programarea Funcțională</b> <ul style="list-style-type: none"> <li>• Funcții pure, imutabilitate.</li> <li>• Funcții lambda în Python.</li> </ul>	2		
<b>7: Aplicații în Programarea Funcțională</b> <ul style="list-style-type: none"> <li>• Funcții de ordin superior: map.</li> </ul>	2		
<b>8: Programarea Logică – Introducere</b> <ul style="list-style-type: none"> <li>• Ce este programarea logică?</li> </ul>	2		
<b>9: Aplicații în Programarea Logică</b> <ul style="list-style-type: none"> <li>• Scrierea de interogări simple.</li> </ul>	2		
<b>10: Programarea Concurrentă – Introducere</b> <ul style="list-style-type: none"> <li>• Concepte de bază: fire de execuție, procese.</li> <li>• Exemple simple în Python.</li> </ul>	2		
<b>11: Aplicații Concurente</b> <ul style="list-style-type: none"> <li>• Sincronizarea firelor de execuție.</li> </ul>	2		
<b>12: Paradigma Evenimentială</b> <ul style="list-style-type: none"> <li>• Programarea bazată pe evenimente: GUI simplu.</li> </ul>	2		
<b>13: Revizuire și Recapitulare</b> <ul style="list-style-type: none"> <li>• Recapitulare a conceptelor învățate.</li> </ul>	2		
<b>14: Evaluare finală</b> <ul style="list-style-type: none"> <li>• Prezentare subiect model examen.</li> </ul>			
<b>Bibliografie</b>			
<ul style="list-style-type: none"> <li>• Programare logica si programare functionala. Teorie si aplicatii, Mihaela Oprea, ISBN: 9786062505509.</li> <li>• Harrison, J. (2009). Handbook of Practical Logic and Automated Reasoning. Cambridge University Press.</li> <li>• Genesereth, M. (2010). CS157: Computational Logic, curs Stanford. <a href="http://logic.stanford.edu/classes/cs157/2010/cs157.html">http://logic.stanford.edu/classes/cs157/2010/cs157.html</a></li> <li>• Curs de programare in Python3 - Structuri de date si algoritmi, Vlad Tudor, ISBN: 9786306559015</li> </ul>			

Bibliografie minimală
<ul style="list-style-type: none"> <li>• <a href="https://www.w3schools.com/python/">https://www.w3schools.com/python/</a></li> <li>• <a href="https://verificationguide.com/systemverilog/systemverilog-tutorial/">https://verificationguide.com/systemverilog/systemverilog-tutorial/</a> task-uri și procese.</li> </ul>

Aplicații (Seminar / laborator / lucrări practice / proiect)	Nr. ore	Metode de predare	Observații
<b>1: Configurare și Introducere</b> <ul style="list-style-type: none"> <li>• Instalarea mediilor de lucru (Python, IDE-uri).</li> <li>• Primul program în Python.</li> </ul>	2	conversația, lucrări practice, exercițiul, problematizarea, proiectul	
<b>2: Exerciții de Bază în Programarea Imperativă</b> <ul style="list-style-type: none"> <li>• Scrierea de programe simple (bucle, condiționale).</li> </ul>	2		
<b>3: Funcții în Python</b> <ul style="list-style-type: none"> <li>• Definirea și utilizarea funcțiilor.</li> </ul>	2		
<b>4: Introducere în OOP</b> <ul style="list-style-type: none"> <li>• Crearea de clase și obiecte.</li> </ul>	2		
<b>5: Aplicații OOP</b> <ul style="list-style-type: none"> <li>• Proiect simplu folosind clase.</li> </ul>	2		
<b>6: Programarea Funcțională – Exerciții Simple</b> <ul style="list-style-type: none"> <li>• Utilizarea funcțiilor lambda și a funcțiilor de ordin superior.</li> </ul>	2		
<b>7: Aplicații Funcționale</b> <ul style="list-style-type: none"> <li>• Procesarea listelor cu map.</li> </ul>	2		
<b>8: Programare Concurrentă – Introducere</b> <ul style="list-style-type: none"> <li>• Concepte de bază: fire de execuție (threads).</li> <li>• Crearea de programe concurente simple în Python.</li> </ul>	2		
<b>9: Aplicații în Programarea Concurrentă</b> <ul style="list-style-type: none"> <li>• Sincronizarea firelor de execuție.</li> <li>• Exerciții practice cu fire de execuție pentru probleme simple.</li> </ul>	2		
<b>10: Programarea Evenimentială</b> <ul style="list-style-type: none"> <li>• Crearea unor aplicații interactive: GUI de bază în Python (Tkinter).</li> </ul>	2		
<b>11: Proiect Concurrent</b> <ul style="list-style-type: none"> <li>• Proiect simplu folosind fire de execuție.</li> </ul>	2		
<b>12: Aplicații Complexe în OOP</b> <ul style="list-style-type: none"> <li>• Dezvoltarea unui proiect care combină OOP și programarea funcțională.</li> </ul>	2		
<b>13: Proiect Final – Aplicație Multi-Paradigmă</b> <ul style="list-style-type: none"> <li>• Dezvoltarea unei aplicații simple care să combine mai multe paradigme.</li> </ul>	2		
<b>14: Prezentarea Proiectelor</b> <ul style="list-style-type: none"> <li>• Prezentarea proiectelor realizate.</li> </ul>	2		

Bibliografie
<ul style="list-style-type: none"> <li>• Curs de programare in Python3 - Structuri de date si algoritmi, Vlad Tudor, ISBN: 9786306559015</li> <li>• <a href="https://www.pythonisti.ro/">https://www.pythonisti.ro/</a></li> </ul>
Bibliografie minimală
<ul style="list-style-type: none"> <li>• <a href="https://www.tutorialspoint.com/python/index.htm">https://www.tutorialspoint.com/python/index.htm</a></li> <li>• <a href="https://www.w3schools.com/python/">https://www.w3schools.com/python/</a></li> </ul>

**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>• Conținutul cursului și al laboratorului;</li> <li>• vine în întâmpinarea solicitărilor unor discipline ulterioare din planul de învățământ;</li> <li>• este compatibil cu cursuri similare din țară;</li> <li>• pregătește viitorii absolvenți pentru a preda discipline specifice TIC</li> </ul>
--

**10. Evaluare**

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Conform cu obiectivele specifice ale disciplinei	conversație evaluatoare	50%

	și prin raportare la conținuturile specifice cursurilor.		
Seminar			
Laborator/lucrări practice	Conform cu obiectivele specifice ale disciplinei și prin raportare la conținuturile specifice activităților practice.	probă practică	50%
Proiect			

10.1. Standard minim de performanță evaluare la curs	
<ul style="list-style-type: none"> <li>• Înțelegerea și aplicarea corectă a paradigmelor de bază,</li> <li>• Capacitatea de a analiza codul și a identifica paradigma utilizată.</li> </ul>	
10.2. Standard minim de performanță evaluare la activitatea aplicativă	
<ul style="list-style-type: none"> <li>• Implementarea unor programe simple utilizând diferite paradigm,</li> <li>• Rezolvarea de probleme practice cu ajutorul conceptelor învățate.</li> </ul>	

Data completării	Semnătura titularului de curs	Semnătura cadrului didactic coordonator
<b>23.09.2024</b>		

Data avizării	Semnătura responsabilului de program
<b>24.09.2024</b>	

Data avizării în departament	Semnătura directorului de departament
<b>25.09.2024</b>	

Data aprobării în consiliul facultății	Semnătura decanului
<b>27.09.2024</b>	