

FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Facultatea de Inginerie Electrică și Știința Calculatoarelor
Departamentul	Departamentul de Calculatoare
Domeniul de studii	Calculatoare și tehnologia informației
Ciclul de studii	Program postuniversitar
Programul de studii	Program de conversie profesională – Tehnologia informației și comunicațiilor

2. Date despre disciplină

Denumirea disciplinei	PARADIGME DE PROGRAMARE				
Anul de studiu	II	Semestrul	3	Tipul de evaluare	E
Regimul disciplinei	Categorია formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DS
	Categorია de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă				DOP

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	4	Curs	2	Seminar	0	Laborator/lucrări practice	2	Proiect	0
I b) Totalul de ore pe semestru din planul de învățământ	56	Curs	28	Seminar	0	Laborator/lucrări practice	28	Proiect	0

Distribuția fondului de timp pe semestru	ore
II.a) Studiu individual	116
II.b) Tutoriat (pentru ID)	
III. Examinări	3
IV. Alte activități (precizați):	

Total ore studiu individual (II.a+II.b+III)	116
Total ore pe semestru (I.b+II.a+II.b+III+IV)	175
Numărul de credite	7

4. Competențe specifice acumulate

Competențe profesionale/generale	C1. Operarea cu fundamente științifice și ale informaticii, C2. Proiectarea componentelor software, C3. Soluționarea problemelor foloșind instrumentele tehnologiei informației și comunicațiilor, C4. Îmbunătățirea performanțelor sistemelor informatice.
Competențe transversale	CT1. Comportarea onorabilă, respectabilă, etică, în spiritul legii pentru a asigura reputația profesiei, CT3. Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale și de cultură organizațională.

5. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

Obiectivul general al disciplinei	Înțelegerea fundamentelor matematice ale TIC și oferirea unei perspective de ansamblu asupra TIC.
-----------------------------------	---

6. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1: Introducere în Paradigmele de Programare - Ce este o paradigmă de programare? - Principalele paradigme (imperativă, orientată pe obiecte, funcțională, logică). - Exemple de limbaje de programare.	2	expunerea, prelegerea, dezbateri, problematizarea, conversația, demonstrația	
2: Programarea Imperativă - Ce este programarea imperativă? - Structuri de control: bucle, condiționale. - Exemple simple în Python.	2		

3: Programarea Procedurală - Organizarea codului în funcții. - Parametri și valori returnate. - Aplicații simple cu funcții în Python.	2		
4: Programarea Orientată pe Obiecte – Partea 1 - Noțiuni de bază: clase și obiecte. - Exemple simple de clase și obiecte în Python.	2		
5: Programarea Orientată pe Obiecte – Partea 2 - Moștenirea și polimorfismul. - Exemple practice.	2		
6: Introducere în Programarea Funcțională - Funcții pure, imutabilitate. - Funcții lambda în Python.	2		
7: Aplicații în Programarea Funcțională Funcții de ordin superior: map.	2		
8: Programarea Logică – Introducere Ce este programarea logică?	2		
9: Aplicații în Programarea Logică Scrierea de interogări simple.	2		
10: Programarea Concurrentă – Introducere - Concepte de bază: fire de execuție, procese. - Exemple simple în Python.	2		
11: Aplicații Concurente Sincronizarea firelor de execuție.	2		
12: Paradigma Evenimentială Programarea bazată pe evenimente: GUI simplu.	2		
13: Revizuire și Recapitulare Recapitulare a conceptelor învățate.	2		
14: Evaluare finală Prezentare subiect model examen.			
Bibliografie minimală recomandată			
https://www.w3schools.com/python/ https://verificationguide.com/systemverilog/systemverilog-tutorial/ task-uri și procese.			

Aplicații (Seminar / laborator / lucrări practice / proiect)	Nr. ore	Metode de predare	Observații
1: Configurare și Introducere - Instalarea mediilor de lucru (Python, IDE-uri). - Primul program în Python.	2	conversația, lucrări practice, exercițiul, problematizarea, proiectul	
2: Exerciții de Bază în Programarea Imperativă Scrierea de programe simple (bucle, condiționale).	2		
3: Funcții în Python Definirea și utilizarea funcțiilor.	2		
4: Introducere în OOP Crearea de clase și obiecte.	2		
5: Aplicații OOP Proiect simplu folosind clase.	2		
6: Programarea Funcțională – Exerciții Simple Utilizarea funcțiilor lambda și a funcțiilor de ordin superior.	2		
7: Aplicații Funcționale Procesarea listelor cu map.	2		
8: Programare Concurrentă – Introducere - Concepte de bază: fire de execuție (threads). - Crearea de programe concurente simple în Python.	2		
9: Aplicații în Programarea Concurrentă - Sincronizarea firelor de execuție. - Exerciții practice cu fire de execuție pentru probleme simple.	2		
10: Programarea Evenimentială Crearea unor aplicații interactive: GUI de bază în Python (Tkinter).	2		
11: Proiect Concurrent Proiect simplu folosind fire de execuție.	2		
12: Aplicații Complexe în OOP	2		

Dezvoltarea unui proiect care combină OOP și programarea funcțională.			
13: Proiect Final – Aplicație Multi-Paradigmă Dezvoltarea unei aplicații simple care să combine mai multe paradigme.	2		
14: Prezentarea Proiectelor Prezentarea proiectelor realizate.	2		
Bibliografie minimală recomandată			
https://www.tutorialspoint.com/python/index.htm https://www.w3schools.com/python/			

7. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Conform cu obiectivele specifice ale disciplinei și prin raportare la conținuturile specifice cursurilor.	conversație evaluatoare	50%
Seminar			
Laborator/lucrări practice	Conform cu obiectivele specifice ale disciplinei și prin raportare la conținuturile specifice activităților practice.	probă practică	50%
Proiect			

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
24.09.2025	ș.l.dr.ing. ȘIEAN Alexandru-Ionuț	ș.l.dr.ing. ȘIEAN Alexandru-Ionuț

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
25.09.2025	ș.l. dr. inf. BĂRÎLĂ Adina-Luminița

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
25.09.2025	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
26.09.2025	Prof. dr. ing. Laurentiu- Dan MILICI