FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Facultatea de Inginerie Electrică şi Ştiinţa Calculatoarelor
Departamentul	Departamentul de Calculatoare, Electronică și Automatică
Domeniul de studii	Inginerie electronică, telecomunicații și tehnologii informaționale
Ciclul de studii	Licență
Programul de studii	Rețele și software de telecomunicații / Electronică aplicată

2. Date despre disciplină

Denumirea disciplinei	**INFORMATICĂ APLICATĂ**				
Anul de studiu	I	Semestrul	2	Tipul de evaluare	E
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DF
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă				DOB

3. Timpul total estimat (ore alocate activităţilor 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	

Distribuția fondului de timp pe semestru	ore
II. Studiu individual	41
III. Examinări	3
IV. Alte activități (precizați): Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	

Total ore studiu individual (II +III)	44
Total ore pe semestru (I.b+II +III+IV)	100
Numărul de credite	4

4. Competențe specifice acumulate

Competențe transversale	**CT 1** - Efectueaza calcule
Competente profesionale	**CP 7** - Identifica nevoile clientilor **CP 8** - Executa calcule matematice analitice **CP 9** - Stabileste procese de date **CP 21** Aplica tehnici de analiza statistica

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul/absolventul identifică și descrie conceptele, principiile și metodele de bază ale programării în limbajul Python (tipuri de date, structuri de control, funcții, module, structuri de date fundamentale, lucrul cu fișiere) și metodele numerice și de prelucrare a datelor utilizate în rezolvarea problemelor de matematică și fizică cu aplicabilitate în electronică.	Studentul/absolventul operează cu concepte și metode de bază din matematică, fizică și informatică pentru a specifica cerințe, a elabora, a implementa și a testa programe modulare în limbajul Python, folosind biblioteci specifice (NumPy, Matplotlib, SciPy) și bune practici de programare. Studentul/absolventul rezolvă probleme de matematică și fizică cu aplicabilitate în inginerie electronică (calcule de circuite, prelucrarea și reprezentarea grafică a	Studentul/absolventul este angajat în învățarea pe tot parcursul vieții pentru dobândirea și implementarea cunoștințelor, după cum este necesar, folosind strategii de învățare adecvate Studentul/absolventul este angajat în învățarea pe tot parcursul vieții pentru dobândirea și implementarea cunoștințelor, după cum este necesar, folosind strategii de învățare adecvate.

	datelor de măsurare și a semnalelor, comunicația cu echipamente și plăci de dezvoltare), interpretează rezultatele și validează soluția obținută.	

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

Obiectivul general al disciplinei	Dezvoltarea competențelor practice și teoretice necesare pentru utilizarea limbajului Python în proiectarea, implementarea și testarea aplicațiilor informatice destinate procesării datelor și controlului echipamentelor și sistemelor medicale, printr-o abordare modularizată și orientată spre integrarea software-hardware.

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
I. Introducere în limbajul Python		expunere orală, conversație, exemplu demonstrativ, sinteza cunoștințelor, descoperire dirijată, exemplificare, studiu de caz	
1. Caracteristici generale ale limbajului Python. Compilator vs. interpretor. Variabile. Tipuri de date. Operatori.	3		
2. Structuri de control în Python. Sintaxa generala. Particularități. Exemple. Aplicații	3		
3. Manipularea șirurilor de caractere în Python	3		
II. Structuri de date în Python.			
1. Tipuri de structuri de date (colecții). 2. Liste în Python. Vectori si matrice manipulați prin liste. 3. Tuple. Mulțimi. Dicționare	4		
III.Programare procedurală.			
1. Definirea funcțiilor in Python. 2. Funcții cu număr variabil de argumente (împachetarea și despachetarea argumentelor). 3. Tratarea erorilor: excepții, try/except, validarea datelor de intrare. Testarea programelor (assert, cazuri de test) și depanarea.	3		
IV.Module, fișiere și elemente de programare orientată pe obiecte			
1. Module și pachete proprii. Biblioteca standard (math, time, datetime, random). Instalarea pachetelor cu pip și mediile virtuale. 2. Operații de I/O cu fișiere text, binare, CSV și JSON. Managerul de context *with*.	4		
V. Procesarea si vizualizarea datelor utilizând limbajul Python			
1. NumPy: tablouri, operații vectoriale, algebră liniară. 2. Procesarea datelor de măsurare: citire CSV, statistici, filtrare (cu NumPy sau pandas, la nivel introductiv). 3. Matplotlib	4		
VI. Folosirea instrumentelor de dezvoltare Python pentru dezvoltarea proiectelor hardware			
1. Instrumente Python pentru controlul echipamentelor hardware si aplicații demonstrative.	4		

Bibliografie minimală recomandată

1. Luciano Ramalho, Fluent Python. Clear, Concise, and Effective Programming, O'Reilly Media, August 2015.
2. Python Software Foundation - https://www.python.org/, versiune electronică, 2025
3. Tutoriale Python: http://docs.python.org/tutorial/index.html, https://www.codecademy.com/learn/learn-python, versiune electronică, 2023
4. MicroPython – Python 3 for microcontrollers - https://micropython.org/
5. Python Resources for NI Hardware and Software - https://www.ni.com/ro-ro/support/documentation/supplemental/16/python-resources-for-ni-hardware-and-software.html
6. PyVISA: Control your instruments with Python - https://pyvisa.readthedocs.io/en/latest/

Aplicații (Seminar / **laborator** / lucrări practice / proiect)	Nr. ore	Metode de predare	Observații
1. Norme de protecția muncii. Introducere în Python. Prezentarea mediului de dezvoltare Python	2	lucrări practice, dezvoltarea de programe pe calculator, experimentul	În cadrul lucrărilor de laborator se urmăreşte ca programele realizate de studenţi să respecte o serie de reguli de bază ale ingineriei software. Temele de casă vor consta din dezvoltarea unor programe de complexitate medie, crearea documentaţiei aferente produsului program realizat. Etape principale în dezvoltarea temei: ● însuşirea temei de realizat; ● proiectarea logică a produsului program; ● stabilirea structurilor de date şi a algoritmului; ● inserarea de comentarii în codul sursă ● punerea la punct a produsului program; ● testarea cu date impuse.
2. Structuri de control în Python (I)	2		
3. Structuri de control în Python (II)	2		
3. Manipularea șirurilor de caractere în Python	2		
5. Aplicații folosind colecții în Python (liste, tuple)	2		
6. Test de evaluare (1)	2		
7. Dicționare în Python	2		
8. Funcții în Python	2		
9. Operații cu fișiere text.	2		
10. NumPy	2		
11. Matplotlib: vizualizarea datelor de măsurare	2		
12. Control hardware utilizând limbajul Python (I)	2		
13. Control hardware utilizând limbajul Python (II)	2		
14. Test de evaluare (2)	2		

Bibliografie minimală recomandată

1. Luciano Ramalho, Fluent Python. Clear, Concise, and Effective Programming, O'Reilly Media, August 2015.
2. Rance D. Necaise, Data Structures and Algorithms Using Python, Ed. JOHN WILEY & SONS, 2011.
3. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Data Structures and Algorithms in Python, John Wiley & Sons, 2013.
4. Python Software Foundation - https://www.python.org/, versiune electronică, 2023
5. The Python standard library. http://docs.python.org/py3k/library/index.html, versiune electronică, 2022
7. Tutoriale Python: http://docs.python.org/tutorial/index.html, https://www.codecademy.com/learn/learn-python, versiune electronică, 2022
8. John V. Guttag, Introduction to Computation and Programming Using Python. The MIT Press (2021)
9. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduction to Algorithms. The MIT Press (2022)

8. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	- Cunoașterea elementelor de bază și a structurilor de date specifice limbajului Python; - Capacitatea de a elabora și implementa un algoritm pentru rezolvarea unei probleme - Capacitatea de utilizare a instrumentelor Python în diverse aplicații.	Evaluare prin probă finală mixta (proba scrisa si proba practica pe calculator)	50%
Evaluare pe parcurs	- Capacitatea de implementare a unui program în limbajul Python. - Capacitatea de a utiliza resurse Python pentru îndeplinirea unor cerințe impuse.	Evaluare continuă (prin metode orale şi probe practice)	50%

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
23.09.2026	Ș.l. dr. ing. Ionela RUSU	As. univ. drd. ing. Alexandru Tudor ANDREI

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
24.09.2026	Conf. dr. ing. Eugen COCA

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Conf. dr. ing. Eugen COCA

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof. dr. ing. Laurentiu-Dan MILICI