

## FIȘA DISCIPLINEI

(licență)

### 1. Date despre program

Instituția de învățământ superior	Universitatea „Ștefan cel Mare” din Suceava
Facultatea	Facultatea de Inginerie Electrică și Știința 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	<b>INFORMATICĂ APLICATĂ</b>				
Titularul activităților de curs	Ș.l. dr. ing. Ionela RUSU				
Titularul activităților aplicative	Ș.l. dr. ing. Dragoș VICOVEANU				
Anul de studiu	I	Semestrul	2	Tipul de evaluare	E
Regimul disciplinei	Categorია formativă a disciplinei DF - fundamentală, DD - în domeniu, DS - de specialitate, DC – complementară				DF
	Categorია de opționalitate a disciplinei: DI - impusă, DO - opțională, DF - facultativă				DI

### 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	14
II b) Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	6
II c) Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	21
II d) Tutoriat	
III Examinări	3
IV Alte activități (precizați):	

Total ore studiu individual II (a+b+c+d)	41
Total ore pe semestru (Ib+II+III+IV)	100
Numărul de credite	4

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

Curriculum	
Competențe	

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

Desfășurare a cursului		<ul style="list-style-type: none"> <li>PC, video proiector, exemple de programe funcționale pe calculator, suport de curs în format electronic, conexiune la internet</li> </ul>
Desfășurare aplicații	Seminar	<ul style="list-style-type: none"> <li></li> </ul>
	Laborator/lucrări practice	<ul style="list-style-type: none"> <li>Laborator dotat cu minim 14 calculatoare având instalat mediul de dezvoltare pentru limbajul Python, ghid de lucrări practice în format electronic, conexiune la internet</li> </ul>
	Proiect	<ul style="list-style-type: none"> <li></li> </ul>

## 6. Competențe specifice acumulate

Competențe profesionale	C3. Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare C4. Proiectarea și utilizarea unor aplicații hardware și software de complexitate redusă specifice electronicii aplicate
Competențe transversale	

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

Obiectivul general al disciplinei	<ul style="list-style-type: none"> <li>• Însușirea unor aspecte complexe ale limbajului Python și creșterea capacității de dezvoltarea sistematică a programelor</li> <li>• Dezvoltarea deprinderilor de proiectare structurată a programelor și aprofundarea noțiunilor de tipuri structurate de date;</li> <li>• Dezvoltarea capacităților de implementare modularizată a produselor program;</li> <li>• Dobândirea cunoștințelor în utilizarea de instrumente software pentru procesarea datelor și dezvoltarea aplicațiilor hardware.</li> </ul>
-----------------------------------	--

## 8. Conținuturi

Curs	Nr. ore	Metode de predare	Observații	
<b>I. Introducere în limbajul Python</b>				
1. Caracteristici generale ale limbajului Python. Compilator vs. interpretor. Variabile. Tipuri de date. Operatori.	3	expunere orală, conversație, exemplu demonstrativ, sinteza cunoștințelor, descoperire dirijată, exemplificare, studiu de caz		
2. Structuri de control în Python. Sintaxa generală. Particularități. Exemple. Aplicații	3			
3. Manipularea șirurilor de caractere în Python	3			
<b>II. Structuri de date în Python.</b>				
1. Tipuri de structuri de date (colecții).	4			
2. Liste în Python. Vectori și matrice manipulați prin liste.				
3. Tuple. Mulțimi. Dicționare				
<b>III. Programare procedurală.</b>				
1. Definirea funcțiilor în Python.	3			
2. Funcții cu număr variabil de argumente (împachetarea și despachetarea argumentelor).				
3. Aplicații.				
<b>IV. Programare modulară.</b>				
1. Operații I/O în Python.	4			
2. Lucrul cu module Python.				
3. Operații de I/O cu fișiere text, binare, CSV				
<b>V. Procesarea și vizualizarea datelor utilizând limbajul Python</b>				
1. Instrumente Python pentru procesarea datelor	2			
2. Instrumente Python pentru vizualizarea datelor	2			
<b>VI. Folosirea instrumentelor de dezvoltare Python pentru dezvoltarea proiectelor hardware</b>				
1. Instrumente Python pentru controlul echipamentelor hardware.	4			
2. Aplicații demonstrative.				

### Bibliografie

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, 2024
5. The Python language reference. <http://docs.python.org/py3k/reference/index.html>, versiune electronică, 2024
6. The Python standard library. <http://docs.python.org/py3k/library/index.html>, versiune electronică, 2024
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)

10. Mark Lutz. Learning Python. O'Reilly. 4<sup>th</sup> Edition. 2009

11. MicroPython – Python 3 for microcontrollers - <https://micropython.org/>

12. 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>

13. PyVISA: Control your instruments with Python - <https://pyvisa.readthedocs.io/en/latest/>

**Bibliografie minimală**

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, 2024

Aplicații (Seminar / <b>laborator</b> / 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 urmărește ca programele realizate de studenți să respecte o serie de reguli de bază ale ingineriei software. Temele de casă vor consta din dezvoltarea unor programe de complexitate medie, crearea documentației aferente produsului program realizat. Etapă principale în dezvoltarea temei: <ul style="list-style-type: none"> <li>• însușirea temei de realizat;</li> <li>• proiectarea logică a produsului program;</li> <li>• stabilirea structurilor de date și a algoritmului;</li> <li>• inserarea de comentarii în codul sursă</li> <li>• punerea la punct a produsului program;</li> <li>• testarea cu date impuse.</li> </ul>
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. Utilizarea colecțiilor iterative pentru manipularea volumelor mari de date	2		
11. Utilizare module Python pentru procesarea datelor	2		
12. Control hardware utilizând limbajul Python (I)	2		
13. Control hardware utilizând limbajul Python (II)	2		
14. Test de evaluare (2)	2		

Prezența la activitățile obligatorii este reglementată de “R12. Regulamentul cadru privind evaluarea studenților” ([https://usv.ro/fisiere\\_utilizator/file/ceac/regulamente/R12%20Reg%20cadru%20privind%20evaluarea%20stud\\_Ed%204rev2\\_25.05.2023\\_site.pdf](https://usv.ro/fisiere_utilizator/file/ceac/regulamente/R12%20Reg%20cadru%20privind%20evaluarea%20stud_Ed%204rev2_25.05.2023_site.pdf)). Recuperarea activităților aplicative obligatorii programate pe parcursul semestrului se face în conformitate cu precizările aceluiași regulament. De asemenea, în conformitate cu prevederile articolului 5.2.3, activitățile pe parcurs pot fi echivalate cu proiecte din tematica disciplinei cu acordul cadrului didactic titular.

**Bibliografie**

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)

**Bibliografie minimală**

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, 2022

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>• Limbajul Python se află în pe primul loc în topul limbajelor de programare conform The TIOBE Programming Community index (septembrie 2022) și IEEE Spectrum (septembrie 2022), limbajul Python înregistrând o creștere constantă a popularității în ultimii ani.</li> <li>• Cursul oferă cunoștințele de programare medii spre avansate în limbajul Python necesare pentru rezolvarea problemelor evidențiate de potențialii angajatori.</li> <li>• <i>Compatibilitate națională.</i> Conținut similar cu cursul de Structuri de date din anul I - AIA UPB, Structuri de date și algoritmi, an 2 AIA – Universitatea Politehnica Timisoara, Programarea calculatoarelor IC UT Iasi, Tehnici de programare, anul IC, U.P. Timisoara, programarea calculatoarelor-tehnici de programare, anul IC, Universitatea Craiova.</li> <li>• <i>Compatibilitate internațională:</i> <ul style="list-style-type: none"> <li>- Harvard University (SUA): CS50 - Introduction to Programming with Python;</li> <li>- Massachusetts Institute of Technology (SUA): 6.006 Introduction to Algorithms, 6.100A/B Introduction to CS and Programming using Python;</li> <li>- Stanford School of Engineering (SUA): CS106A - Programming Methodology, CS106B -Programming Abstractions, XCME009 - Introduction to Python, CS106X - Programming Abstractions (Accelerated).</li> </ul> </li> </ul>
---

10. **Evaluare**

10.1. Standard minim de performanță evaluare la curs

10.2. Standard minim de performanță evaluare la activitatea aplicativă

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	<ul style="list-style-type: none"> <li>- Cunoașterea elementelor de bază și a structurilor de date specifice limbajului Python;</li> <li>- Capacitatea de a elabora și implementa un algoritm pentru rezolvarea unei probleme</li> <li>- Capacitatea de utilizare a instrumentelor Python în diverse aplicații.</li> </ul>	Evaluare prin probă finală mixta (proba scrisă și proba practică pe calculator)	50%
Seminar			
Laborator / lucrări practice	<ul style="list-style-type: none"> <li>- Capacitatea de implementare a unui program în limbajul Python.</li> <li>- Capacitatea de a utiliza resurse Python pentru îndeplinirea unor cerințe impuse.</li> </ul>	Evaluare continuă (prin metode orale și probe practice)	50%
Proiect			

Standard minim de performanță

Standard minim de performanță evaluare la curs

- rezolvarea unor probleme simple în limbajul Python;
- o bună descompunere a problemei în subprobleme mai simple;
- asigurarea de comentarii suficiente permițând o întreținere facilă a programului realizat.

Standard minim de performanță evaluare la activitatea aplicativă:

- adaptarea unei concepții modulare în elaborarea programelor;
- reutilizarea unor module deja testate;
- utilizarea de instrumente Python în dezvoltarea de aplicații.

Data completării	Semnătura titularului de curs	Semnătura titularului de aplicație
19.09.2024		

Data avizării	Semnătura responsabilului de program
20.09.2024	

Data avizării în departament	Semnătura directorului de departament
23.09.2024	

Data aprobării în consiliul facultății	Semnătura decanului
27.09.2024	