Ministerul Educației și Cercetării
Universitatea Valahia din Târgoviște
Facultatea de Inginerie Electrică, Electronică și Tehnologia Informației
Departamentul de Electronică, Telecomunicații și Inginerie Energetică

# FIŞA DISCIPLINEI

## 1. Date despre program

| | |
|---|---|
| 1.1 Instituția de învățământ superior | Universitatea VALAHIA din Târgoviște |
| 1.2 Facultatea/Departamentul | Inginerie Electrică, Electronică şi Tehnologia Informaţiei |
| 1.3 Departamentul | Electronică, Telecomunicaţii şi Inginerie Energetică |
| 1.4 Domeniul de studii | Inginerie Electronică, Telecomunicații și Tehnologii Informaționale |
| 1.5 Ciclul de studii | Licență |
| 1.6 Programul de studii/Calificarea | Electronică aplicată |

## 2. Date despre disciplină

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2.1 Denumirea disciplinei | | | | **Programare în Python** | | | |
| 2.2 Titularul activităților de curs | | | | Șl. univ. dr. ing. Ion CĂCIULĂ | | | |
| 2.3 Titularul activităților de seminar | | | | Șl. univ. dr. ing. Ion CĂCIULĂ | | | |
| 2.4 Anul de studiu | II | 2.5 Semestrul | I | 2.6 Tipul de evaluare | C | 2.7 Regimul disciplinei | B S |

## 3. Timpul total estimat (ore pe semestru al activităților didactice)

| | | | | | |
|---|---|---|---|---|---|
| 3.1 Număr de ore pe săptămână | 3 | din care: 3.2 curs | 1 | 3.3 seminar/laborator | 2 |
| 3.4 Total ore din planul de învățământ | 42 | din care: 3.5 curs | 14 | 3.6 seminar/laborator | 28 |
| Distribuţia fondului de timp | | | | | ore |
| Studiul după manual, suport de curs, bibliografie şi notiţe | | | | | 10 |
| Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate şi pe teren | | | | | 10 |
| Pregătire seminarii/laboratoare, teme, referate, portofolii şi eseuri | | | | | 5 |
| Tutoriat | | | | | 3 |
| Examinări | | | | | 5 |
| Alte activități | | | | | - |
| 3.7 Total ore studiu individual | | | | | 33 |
| 3.9 Total ore pe semestru | | | | | 75 |
| 3.10 Numărul de credite | | | | | 3 |

## 4. Precondiţii (acolo unde este cazul)

| | |
|---|---|
| 4.1 de curriculum | Arhitectura microprocesoarelor, Teoria transmisiunii informaţiei, Programarea calculatoarelor și limbaje, de programare |
| 4.2 de competențe | Cunoștințe generale despre arhitectura microprocesoarelor, variabile, funcții, clase și obiecte |

## 5. Condiţii (acolo unde este cazul)

| | |
|---|---|
| 5.1 de desfășurare a cursului | Videoproiector, tablă inteligentă, tablă albă |
| 5.2 de desfășurare a seminarului/laboratorului | Laboratorul se va desfășura într-o sală dotată cu tablă, videoproiector și calculatoare având instalate mediul de dezvoltare PyCharm, respectiv Spyder |

## **6.** Obiectivele disciplinei (reieşind din grila competenţelor specifice acumulate)

| | |
|---|---|
| 6.1 Obiectivul general al disciplinei | Însușirea conceptelor, principiilor și tehnicilor de programare în limbajul Python și dezvoltarea aptitudinilor de proiectare, implementare, testare și depanare a aplicațiilor software, utilizând structuri de date, funcții, mecanisme de programare orientată pe obiecte și biblioteci specifice, pentru rezolvarea unor probleme de inginerie electronică. |
| 6.2 Obiectivele specifice | - Cunoașterea sintaxei și a elementelor fundamentale ale limbajului Python: variabile, expresii, instrucțiuni, structuri de control și tipuri de date.<br>- Utilizarea structurilor de date, funcțiilor și modulelor pentru organizarea și dezvoltarea aplicațiilor software.<br>- Aplicarea principiilor programării orientate pe obiecte și a mecanismelor de tratare a excepțiilor în dezvoltarea aplicațiilor Python.<br>- Dezvoltarea aptitudinilor de prelucrare a datelor și gestionare a fișierelor prin utilizarea instrumentelor și bibliotecilor specifice.<br>- Realizarea unor aplicații cu interfață grafică, utilizând mediile de dezvoltare și instrumentele disponibile.<br>- Familiarizarea cu tehnicile de programare concurentă și utilizarea bibliotecilor Python pentru execuția paralelă a sarcinilor.<br>- Dezvoltarea capacității de testare, depanare și evaluare a funcționalității aplicațiilor realizate. |

## **7.** Rezultatele învățării

| |
|---|
| 7.1 Cunoștințe *(Rezultatul asimilării de informații prin învățare. Cunoștințele reprezintă ansamblul de fapte, principii, teorii și practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice și/sau faptice)*<br>- C1. Studentul/absolventul descrie și explică elementele fundamentale ale limbajului Python, structurile de date și de control, funcțiile, principiile programării orientate pe obiecte, mecanismele de tratare a excepțiilor, operațiile cu fișiere și tehnicile de programare concurentă. |
| 7.2 Aptitudini *(Capacitatea de a aplica cunoștințe și de a utiliza know-how pentru a duce la îndeplinire sarcini și a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive și creative) sau practice (implicând dexteritate manuală și utilizarea de metode, materiale, unelte și instrumente))*<br>- A1. Studentul/absolventul dezvoltă aplicații software în Python utilizând variabile, structuri de control, colecții de date, funcții și module pentru rezolvarea unor probleme de programare.<br>- A2. Studentul/absolventul analizează cerințele unor probleme inginerești și implementează soluții software utilizând programarea orientată pe obiecte, gestionarea fișierelor și tratarea excepțiilor.<br>- A3. Studentul/absolventul proiectează și implementează aplicații Python cu interfață grafică, utilizând biblioteci și instrumente specifice pentru introducerea, prelucrarea și afișarea datelor.<br>- A4. Studentul/absolventul utilizează biblioteci și instrumente Python pentru programare concurentă, testează și depanează aplicațiile dezvoltate și interpretează rezultatele execuției acestora. |
| 7.3 Responsabilitate și autonomie *(Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale)*<br>- RA1. Studentul/absolventul își asumă responsabilitatea pentru corectitudinea aplicațiilor dezvoltate, utilizarea etică a datelor și respectarea principiilor integrității academice.<br>- RA2. Studentul/absolventul demonstrează autonomie în utilizarea documentației tehnice, în identificarea și corectarea erorilor și în adaptarea la noi biblioteci și instrumente de programare.<br>- RA3. Studentul/absolventul colaborează eficient în activitățile de laborator, comunică soluțiile implementate și argumentează alegerile tehnice realizate. |
| 7.4 Arii de învățare EUR-ACE<br>- CI – Cunoaștere și înțelegere: cunoașterea principiilor programării în Python, a structurilor de date, a programării orientate pe obiecte și a mecanismelor de execuție și gestionare a aplicațiilor software.<br>- AI – Analiză inginerească: analiza problemelor de programare, identificarea soluțiilor algoritmice adecvate și evaluarea funcționalității aplicațiilor dezvoltate.<br>- FO – Formularea de opinii: interpretarea rezultatelor execuției programelor, identificarea erorilor și formularea unor concluzii argumentate privind corectitudinea și eficiența soluțiilor implementate.<br>- CM – Comunicare: prezentarea și explicarea soluțiilor software realizate, utilizarea documentației tehnice și comunicarea rezultatelor activităților practice.<br>- FC – Formare continuă: utilizarea autonomă a documentației, a resurselor educaționale și a bibliotecilor software pentru dezvoltarea și actualizarea cunoștințelor de programare în Python. |

## 8. Conţinuturi

| 8.1 Curs | Metode de predare | Observații |
|---|---|---|
| **Curs** | prelegerea, prelegerea – dezbaterea, explicația, problematizarea, brainstorming-ul, reflecția personală, exercițiul, dezbaterea, studiul de caz<br><br>*Mijloace de învățământ*<br>slide-uri PPT<br>videoproiector<br>laptop | 14h |
| Introducere în Python. Variabile, Expresii și Instrucțiuni. | | 2h |
| Structuri repetitive în Python. Șiruri de caractere | | 2h |
| Liste și Dicționare în Python | | 2h |
| Funcții în Python | | 2h |
| Programare oritentată obiect. Tratarea excepțiilor în Python | | 2h |
| Lucrul cu fișiere în Python. Interfețe grafice în Python | | 2h |
| Programare concurentă în Python | | 2h |
| Bibliografie<br>1. Caciula I. - **Materiale didactice in format electronic pe intranet Valahia** *http://moodle.fie.valahia.ro/*<br>2. Allen B. Downey, Think Python, 3rd Edition, https://allendowney.github.io/ThinkPython/<br>3. RealPython, https://realpython.com/<br>4. Python courses, https://www.datacamp.com/courses/introduction-to-python-for-developers | | |
| 8.2 Laborator | Metode de predare | Observații |
| **Laborator** | | **28h** |
| Introducere în mediul de dezvoltare Pycharm și alte medii de dezvoltare pentru programarea în Python | problematizarea, reflecția personală, exercițiul, dezbaterea, studiul de caz | 2h |
| Programarea structurată folosind Python. Structuri de date în Python | | 4h |
| Funcții în Python. Librăria numba. Librăria multiprocessing | | 6h |
| Programare orientată obiect în Python | | 4h |
| Lucurl cu fișiere în Python. Tratarea excepțiilor | | 4h |
| Interfețe grafice în Python | | 4h |
| Programare concurentă în Python | | 4h |
| Bibliografie<br>1. Caciula I. - **Materiale didactice in format electronic pe intranet Valahia** *http://moodle.fie.valahia.ro/*<br>2. Allen B. Downey, Think Python, 3rd Edition, https://allendowney.github.io/ThinkPython/<br>3. RealPython, https://realpython.com/<br>4. Python courses, https://www.datacamp.com/courses/introduction-to-python-for-developers | | |

## 9. Coroborarea conţinuturilor disciplinei cu aşteptările reprezentanţilor comunităţii epistemice, asociaţiilor profesionale şi angajatori reprezentativi din domeniul aferent programului

Conținuturile disciplinei sunt corelate cu cerințele pregătirii inginerești în domeniul electronicii aplicate, oferind cunoștințe și aptitudini privind dezvoltarea aplicațiilor software în limbajul Python. Studiul structurilor de date, funcțiilor, programării orientate pe obiecte, gestionării fișierelor, interfețelor grafice și programării concurente contribuie la formarea capacității de realizare a soluțiilor informatice utilizate în prelucrarea datelor și automatizarea unor sarcini inginerești. Activitățile de laborator dezvoltă aptitudini de implementare, testare, depanare și evaluare a aplicațiilor software, relevante pentru dezvoltarea și integrarea instrumentelor informatice în sisteme electronice, aplicații de achiziție de date și monitorizare, în concordanță cu cerințele mediului profesional.

## 10. Evaluare

| Tip activitate | 10.1 Criterii de evaluare | 10.2 Metode de evaluare | 10.3 Pondere din nota finală |
|---|---|---|---|
| 10.4 Curs | Cunoașterea și explicarea elementelor limbajului Python, a structurilor de date și de control, funcțiilor, programării orientate pe obiecte, gestionării fișierelor și programării concurente; identificarea soluțiilor adecvate unor probleme de programare. (C1, A1–A2) | Colocviu scris – test grilă | 50% |
| 10.5 Seminar/laborator | Corectitudinea implementării și funcționării aplicației, utilizarea structurilor de date și a bibliotecilor Python, organizarea codului, tratarea erorilor, testarea și depanarea; interpretarea | Realizarea unui teme de casă individuale | 50% |

| | | | |
|---|---|---|---|
| | rezultatelor și justificarea soluției realizate. (A1–A4, RA1–RA3) | | |

10.6 Standard minim de performanță

- Identificarea și explicarea elementelor fundamentale ale limbajului Python, a structurilor de control și de date, a funcțiilor și a principiilor programării orientate pe obiecte. (C1)
- Elaborarea și executarea corectă a unor programe Python simple, utilizând variabile, structuri de control, colecții de date și funcții. (A1)
- Implementarea unei aplicații simple care utilizează clase și obiecte, operații cu fișiere și tratarea excepțiilor. (A2)
- Realizarea, testarea și depanarea unei aplicații Python cu funcționalități de bază, prezentarea rezultatelor și justificarea soluției implementate, cu respectarea principiilor de utilizare responsabilă a resurselor informatice. (A3–A4, RA1–RA3)

Obține minimum 50% din punctajul total cumulat și minimum 50% din punctajul aferent evaluării scrise.

Fișa disciplinei corespunde planului de învățământ care se aplică pentru anul I începând cu anul universitar 2025-2026, fiind valabilă pentru anul universitar 2026-2027

| | | |
|---|---|---|
| Data completării<br>10.09.2026 | Titularul de curs<br>Sl. univ. dr. ing. Ion CĂCIULĂ | Titularul de aplicații<br>Sl. univ. dr. ing. Ion CĂCIULĂ |
| Data avizării în departament<br>23.09.2026 | Director de departament<br>Conf. univ. dr. ing. Dan Constantin PUCHIANU | |
| Data avizării în Consiliul Facultății<br>24.09.2026 | Decan<br>Conf. univ. dr. ing. Nicoleta ANGELESCU | |