Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti
Facultatea de Electronică, Telecomunicaţii şi
Tehnologia Informaţiei

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**
1.2 Facultatea	**Electronică, Telecomunicaţii şi Tehnologia Informaţiei**
1.3 Departamentul	**Electronică Aplicată şi Ingineria Informaţiei**
1.4 Domeniul de studii	Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale
1.5 Ciclul de studii	Licenţă
1.6 Specializarea	Electronică aplicată

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Programarea calculatoarelor şi limbaje de programare 3 - Proiect Python Computer Programming and Programming Languages 3 - Python Project			
2.2 Titularul activităţilor de curs							
2.3 Titularul activităţilor de seminar / laborator				As. drd. ing. Iustin Floroiu/S.l. dr. ing. Liviu Stefan			
2.4 Anul de studiu	2	2.5 Semestrul	1	2.6. Tipul de evaluare	V	2.7 Regimul disciplinei	Op
2.8 Tipul disciplinei		F	2.9 Codul disciplinei	04.F.03.A.009		2.10 Tipul de notare	Nota

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

3.1 Număr de ore pe săptămână	1	Din care: 3.2 curs	0.00	3.3 seminar/laborator	1
3.4 Total ore din planul de învăţământ	14	Din care: 3.5 curs	0	3.6 seminar/laborator	14
Distribuţia fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie şi notiţe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					9
Tutorat					0
Examinări					2
Alte activităţi (dacă există):					0

3.7 Total ore studiu individual	11.00
3.8 Total ore pe semestru	25
3.9 Numărul de credite	1

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Programarea calculatoarelor. Programare obiect orientată.
4.2 de rezultate ale învăţării	Cunoştinţe de bază de instalare programe, utilizare editoare text pentru scriere cod şi realizarea de programe pe calculator.

5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

5.1 Curs	Ședința se va desfășura într-o sală dotată, de preferat, cu videoproiector și acces internet (pentru a permite desfășurare simultană sub formă de videoconferință de tip Teams). Studenții pot lucra pe propriile calculatoare.
5.2 Seminar/ Laborator/Proiect	Prezenţa la şedinţe este obligatorie (conform regulamentului studiilor universitare de licenţă). Studenții pot să lucreze pe propriile computere la ședințe, în săli cu acces la internet.

6. Obiectiv general *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învăţământ al specializării studiate etc.)*

Disciplina are ca obiectiv general familiarizarea studenţilor cu limbajul de programare Python şi tehnologii moderne conexe adesea utilizate în practică.

În orele de proiect, studenţii se vor familiariza cu elemente de bază şi avansate ale limbajului de programare Python, pe calculatorul studentului. Studiul limbajului Python va îmbina predarea de noţiuni cu efectuarea de exerciţii de către student, în mod individual, pe calculatorul personal, identificând şi exersând abilităţile necesare modelării pe calculator a unor situaţii din realitate.

Bazându-se pe cunoştinţele acumulate în urma acestei discipline, viitorul inginer electronist va putea implementa sau modifica programe sau scripturi în linie de comandă specifice activităţilor de dezvoltare şi testare software moderne, scriptabile şi automatizabile, fiind capabil de a realiza programe de la specificarea cerinţelor la execuţia, depanarea şi interpretarea rezultatelor.

7. Competenţe *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

Specifice	**C3.** Aplicarea cunoştinţelor, conceptelor şi metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrollere, limbaje şi tehnici de programare.
Transversale (generale)	**CT1.** Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluţii consacrate, asigurând astfel îndeplinirea sarcinilor profesionale. **CT3.** Adaptarea la noile tehnologii, dezvoltarea profesională şi personală, prin formare continuă folosind surse de documentare tipărite, software specializat şi resurse electronice în limba română şi, cel puţin, într-o limbă de circulaţie internaţională.

8. Rezultatele învăţării *(Sunt enunțuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învățării reflectă realizările studentului și mai puțin intențiile profesorului. Rezultatele învățării informează studenții despre ceea ce se așteaptă de la ei din punct de vedere al performanței, pentru a obține notele și creditele dorite. Sunt definite în termeni concreți, folosind verbe similare exemplelor de mai jos și indică ceea ce se va urmări prin evaluare. Rezultatele învățării vor fi astfel redactate încât să fie evidențiată clar relația față de competențele definite la punctul 7.)*

Cunoştinţe	*Rezultatul asimilării de informaţii prin învăţare. Cunoştinţele reprezintă ansamblul de fapte, principii, teorii şi practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice şi/sau faptice.* Este capabil de a răspunde la o serie de întrebări bazate pe un portofoliu de activităţi efectuate pe parcursul semestrului, ca parte a unei examinări orale. Descrie programul Python necesar pentru a rezolva o problemă dată. Evidenţiază necesitatea utilizării unor module. Descrie concepte din limbajul Python necesare rezolvării unei probleme date.
Aptitudini	*Capacitatea de a aplica cunoştinţe şi de a utiliza know-how pentru a duce la îndeplinire sarcini şi a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive şi creative) sau practice (implicând dexteritate manuală şi utilizarea de metode, materiale, unelte şi instrumente).* Modelează, programează, depanează şi rulează cu succes un program cu scopul obţinerii soluţiei la o problemă dată. Implementează un program cu facilităţi specificate, în limbajul Python. Verifică şi depanează validitatea unui program dat. Operează cu noţiuni legate de tipuri de date de bază (precum conceptul mutabil/imutabil), case sensitive, afişare, structuri de date (liste, şiruri, dicţionare, tupluri etc). Evidenţiază utilitatea operatorului slice. Utilizează instrucţiuni de control flux de execuţie, expresii booleene, funcţii, module, în mod adecvat rezolvării de probleme. Identifică situaţii care beneficiază de utilizarea de noţiuni de programare obiect orientată în Python.
Responsabilitate şi autonomie	*Capacitatea cursantului de a aplica în mod autonom şi responsabil cunoştinţele şi aptitudinile sale.* Demonstrează receptivitate pentru contexte noi de învăţare. Conspectează în avans materialele de proiect, în măsura în care acestea sunt puse la dispoziţie. În caz de absenţă, parcurge pe cont propriu materialul predat, pus la dispoziţie. Rezolvă temele de casă în mod individual, autonom, cu respectarea eticii academice. Respectă principiile de etică academică, realizând individual activităţile marcate în acest sens, citând, de asemenea, corect sursele bibliografice utilizate, dacă situaţia o cere. Aplică principii de etică/deontologie profesională în analiza impactului tehnologic al soluţiilor propuse în domeniul de specialitate asupra mediului înconjurător.

9. Metode de predare *(Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenţilor la stabilirea propriului parcurs de învăţare, cum se identifică eventualele rămâneri în urmă şi ce măsuri remediale se adoptă în astfel de cazuri.)*

Materialele didactice utilizate sunt notele şi prezentările de la şedinţe, disponibile şi în format electronic.

Pornindu-se de analiza caracteristicilor de învăţare ale studenţilor şi de la nevoile lor specifice, procesul de predare va explora metode de predare atât expozitive (prelegerea, expunerea), a problematizării cât şi conversative-interactive, bazate pe modele de învăţare bazate pe acţiune, precum exerciţiul, activităţile practice şi rezolvarea de probleme. Interactivitate cu studenţii prin intermediul părţii aplicative asociate. Sunt rezervate intervale de prezentare, analiză şi rezolvare a unor probleme practice (modelarea realităţii).

La partea aplicativă, predarea se bazează pe folosirea metodei expozitive (acoperind funcţia de comunicare şi demonstrativă). Dialogul este utilizat şi în cadrul şedinţelor aplicative. Acestea sunt necesare pregătirii studenţilor pentru temele de casă şi testele de verificare pe parcurs.

Se va utiliza şi feedback-ul, ca modalitate de adaptare a demersului pedagogic la nevoile de învăţare ale studenţilor.

10. Conţinuturi

PROIECT

Nr. crt.	Conţinutul	Nr. ore
1	Noţiuni introductive. Instalarea Python pe Windows şi/sau Linux. Utilizarea unui interpretor Python. Execuţia de fişiere cu cod Python.	2
2	Tipuri de date fundamentale. Mutabil/imutabil. Denumirea variabilelor. Case sensitive. Afişare. Exerciţii.	2
3	Structuri de date. Liste. Tupluri. Dicţionare. Operatorul slice. Exerciţii.	2
4	Instrucţiuni de control flux de execuţie. Expresii booleene. Funcţii. Module. Exerciţii.	2
5	Noţiuni de programare obiect orientată în Python. Clase. Atribute. Metode. Moştenire. Exerciţii.	2
6	Aplicaţii. Interfeţe vizuale (tip GUI). Interfeţe native prin tkinter. Venv. Interfeţe web prin Flask. Exerciţii.	2
7	Verificare finală	2
	Total:	14

Bibliografie:

V.G. Voiculescu, Proiect Python, materiale suport electronic, https://ctipub.sharepoint.com/:f:/s/ProiectPython-Voiculescu/EoWmoTyZ3idIj26kq8M30pgBFtbztZBxs0O5n5Nl9O-WXA?e=LLStlO
W3schools, Python 3, https://www.w3schools.com/python/, 2024
D. Beazley, B.K. Jones, Python cookbook: Recipes for mastering Python, 3rd edition, O'Reilly, 2013

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare	11.3 Pondere din nota finală
11.4 Curs			
11.5 Seminar/laborator/proiect	Identificarea corectă a contextelor teoretice şi practice de aplicare a conceptelor studiate din Python precum şi a facilităţilor studiate ale acestora. Abilitatea de a modela şi rezolva programatic cu succes o problemă în Python.	Activitatea aplicativă este verificată constant pe tot parcursul semestrului. Teme de casă. Evaluare finală pe bază de proiect.	80%
	Verificarea finală a proiectului şi a documentaţiei	Evaluare finală	20%
11.6 Condiţii de promovare			
Implementarea unui program cu facilităţi specificate în limbajul Python, modelarea, programarea, rularea cu succes în scopul obţinerii soluţiei. Obţinerea a 50% din punctajul total.			

12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)

Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

Python este un limbaj foarte popular, stând la baza unor aplicaţii intens utilizate de social-media, având aplicabilitate şi în domenii de viitor precum inteligenţa artificială, prelucrarea masivelor de date. Suntem într-un proces de adoptare a acestui limbaj de programare la nivelul facultăţii, dar fiecare materie îl descrie doar minimal, în scopul introducerii conceptelor specifice de studiu. Este un limbaj popular şi în unele framework-uri de scriptare, testare, buildare întâlnite în domeniile ţintă ale absolvenţilor noştri, iar cu ajutorul acestei materii viitorul inginer electronist va putea implementa sau modifica programe sau scripturi specifice activităţilor de dezvoltare şi testare software moderne, scriptabile şi automatizabile.

Bazându-se pe cunoştinţele acumulate în urma acestei discipline, viitorul inginer electronist va putea implementa sau modifica programe sau scripturi în linie de comandă specifice activităţilor de dezvoltare şi testare software moderne, scriptabile şi automatizabile, fiind capabil de a realiza programe de la specificarea cerinţelor la execuţia, depanarea şi interpretarea rezultatelor.

Data completării	Titular de curs	Titular(i) de aplicaţii
21.09.2026		As. drd. ing. Iustin Floroiu
		S.l. dr. ing. Liviu Stefan

Data avizării în departament	Director de departament
22.09.2026	Conf. Dr. Bogdan Cristian Florea

Data aprobării în Consiliul Facultăţii	Decan
22.09.2026	Prof. Dr. Mihnea Udrea