

S.A.102 PROGRAMAREA DECLARATIVĂ
1. Date despre disciplină

Facultatea	Calculatoare, Informatică și Microelectronică				
Departamentul	Ingineria Software și Automatică				
Ciclul de studii	Studii superioare de licență, ciclul I				
Programul de studii	0613.3 Inginerie Software				
Anul de studii	Semestrul	Tip de evaluare	Categoria formativă	Categoria de opționalitate	Credite ECTS
Anul IV (<i>învățământ cu frecvență</i>)	7	E	S-Disciplina de specialitate	A - unitate de curs opțională	5

2. Timpul total estimat

Total ore în planul de învățământ		Din care				
		Ore auditoriale			Lucrul individual	
		Curs	Lucrări practice	Seminar	Studiul materialului teoretic	Proiectare
Învățământ cu frecvență	150	45	30		45	30

3. Precondiții de acces la disciplină

Conform planului de învățământ	Studentii trebuie să fi susținut probele de evaluare la PC și POO.
Conform competențelor	Pentru a atinge obiectivele cursului, studenții trebuie să posede cunoștințe despre programare de bază.

4. Condiții de desfășurare a procesului educațional pentru

Curs	Sală dotată cu un proiector. Prezentările sunt plasate pe else.fcim.utm.md , studenții au acces la ele.
Lucrări practice/ seminare	Sală dotată cu calculatoare. Studenții vor perfectă rapoarte conform condițiilor impuse de indicațiile metodice. Termenul de predare a lucrării de laborator – două săptămâni după finalizarea acesteia. Pentru predarea cu întârziere a lucrării aceasta se depunțează cu 1punct.

5. Competențe specifice acumulate

Competențe profesionale	<ol style="list-style-type: none"> 1. Competențe în programarea funcțională: <ul style="list-style-type: none"> ○ Înțelegerea conceptelor și principiilor programării funcționale. ○ Abilitatea de a utiliza funcțiile, recursia și funcțiile de ordin înalt pentru rezolvarea problemelor. ○ Capacitatea de a manipula și transforma datele folosind operații pe liste, dataframe-uri. 2. Competențe în utilizarea limbajului Python: <ul style="list-style-type: none"> ○ Abilitatea de a utiliza Python pentru dezvoltarea de aplicații și soluții software. ○ Cunoașterea și utilizarea librăriilor Python, inclusiv librăriile de machine learning (exemplu: scikit-learn) și Pandas pentru manipularea datelor. 3. Competențe în dezvoltarea de soluții software complexe: <ul style="list-style-type: none"> ○ Capacitatea de a dezvolta soluții software complexe care implică programare funcțională și utilizarea librăriilor de machine learning. ○ Abilitatea de a manipula date mari și complexe cu ajutorul Pandas.
Competențe transversale	<ol style="list-style-type: none"> 1. Realizarea lucrărilor de laborator cu utilizarea corectă a surselor și metodelor specifice, precum și susținerea acestora cu demonstrarea capacității de evaluare calitativă a unor soluții tehnice din domeniu.

2. Abilitatea de a gestiona și utiliza modulele studiate din Python într-un mod eficient, cât și capacitatea de a crea module personalizate pentru funcționalități specifice.

6. Obiectivele disciplinei

Obiectivul general	Scopul cursului este de a introduce studenții în programarea declarativă și să le dezvolte abilitățile în acest domeniu.
Obiectivele specifice	<ol style="list-style-type: none"> 1. Înțelegerea conceptelor de bază ale programării funcționale, cum ar fi funcțiile, recursia și funcțiile de ordin înalt. 2. Dezvoltarea abilităților de lucru cu liste, inclusiv filtrarea, transformarea și agregarea datelor. 3. Explorarea conceptelor avansate de programare funcțională, precum polimorfismul, clasele de tipuri și modularizarea. 4. Utilizarea modulelor avansate în limbajul Python, inclusiv librării de machine learning (exemplu: scikit-learn) și manipularea datelor cu Pandas. 5. Crearea și utilizarea API-urilor în Python, folosind Flask pentru dezvoltarea de servicii web. 6. Înțelegerea programării asincrone în Python cu ajutorul asyncio. 7. Dezvoltarea abilităților de creare și utilizare a modulelor proprii în Python.

7. Conținutul disciplinei

Tematica activităților didactice	învățământ cu frecvență
T1. Noțiuni introductive în Programarea Declarativă . Paradigme de programare. Limbaje funcționale de programare declarativă. T2. Limbajul de programare R. Avantaje și domenii de implementare. Obiecte și operații. R reference card.	3
T3. Analiza datelor și reprezentarea lor graphică (în R). Citirea datelor dintr-un fișier. Salvarea datelor. Generare date, secvențe aleatoare. Gestionarea graficelor.	3
T4. Caracteristici de bază ale limbajului Python. Scrierea și interpretarea codului în Python. Tipuri de date standard. Funcții lambda , map() , filter() , reduce() .	3
T5. Module avansate în Python. Generatori , funcțiile încorporate next() și iter() . Colecții de date. Tuple , namedtuple . Modulele Datetime , Math și Random . Expresii regulate.	3
T6. Colectarea automată a datelor. Web scraping . Obținerea codului html al unei pagini web folosind Requests și Beautiful Soup . Salvarea datelor captate în fișiere .csv. Utilizarea librăriei Pandas pentru importarea și manipularea datelor.	3
T7. Analiza datelor cu Python: librăriile NumPy, Pandas. Crearea listelor cu ajutorul librăriei NumPy . Operații de bază din librăria NumPy. Funcții. Utilizarea librăriei Pandas pentru: serii de date, cadre de date, metoda GroupBy, fuzionarea și concatenarea datelor, diverse operațiuni, date de intrare și ieșire.	3
T8. Vizualizarea datelor folosind Matplotlib și Seaborn . Familiarizarea cu Matplotlib pyplot. Crearea graficelor folosind Matplotlib, Seaborn.	3
T9. Vizualizarea interactivă a datelor folosind librăriile Plotly și Cufflinks . Utilizarea librăriei Cufflinks și funcției iplot() .	3

Tematica activităților didactice	învățământ cu frecvență
T10. Introducere în calcule științifice în Python: SymPy, SciPy. SymPy - Algebră simbolică în Python. SciPy - Biblioteca de algoritmi științifici pentru Python.	3
T11. Prelucrarea imaginilor cu Python. Prezentarea librărilor de bază ce permit manipulări elementare asupra imaginilor: rotire, schimbare rezoluție, aplicare zgomot, reducerea zgomotului cu diverse filtre etc. Analiza exemplurilor.	3
T12. Interfață grafică în Python cu PySimpleGUI . GUI personalizat într-o linie de cod. Exemple de diferite GUI.	3
T13. Prelucrarea Limbajului Natural în Python. Noțiuni generale. Librăria Simpy și NLTK.	3
T14. Tehnici de procesare a limbajului natural utilizând Spacy. Exemple de implementarea a pașilor necesari pentru NLP: tokenizare, lematizare, cuvinte-stop, potrivire șabloane (template matcing) etc.	3
T15. Tehnici de procesare a limbajului natural utilizând NLTK. Recunoașterea sentimentelor din texte (<i>nlTK sentiment vader</i>).	3
T16. Realizarea aplicațiilor web folosind Python și Django .	3
Total curs:	45
P1. Introducere în programarea funcțională. Implementarea funcțiilor de manipulare a datelor (vectori, matrici, liste) într-un limbaj funcțional (R).	4
P2. Tratarea tipurilor complexe de date (vectori, matrici, data frame-uri) în limbajul R. Definirea și apelarea funcțiilor într-un limbaj functional (R). Rezolvarea problemelor utilizând abordarea recursivă.	4
P3. Definirea și apelarea funcțiilor pentru prelucrarea tipurilor complexe de date (data frame-uri). Vizualizarea grafică a rezultatelor în limbajul R.	3
P4. Tratarea, analiza și reprezentarea datelor utilizând funcții R și funcții proprii (lucrare de sinteză).	3
P5. Introducere în Python. Definirea și apelarea funcțiilor de ordin înalt map(), filter(), reduce() și a funcțiilor anonime - lambda()	3
P6. Web Scraping folosind Python (și Beautiful Supp). Extragerea datelor de pe web,	3
P7. Manipularea și curățarea datelor folosind biblioteca Pandas și vizualizarea datele folosind biblioteca Matplotlib (ambele de la Python).	3
P8. Utilizarea librărilor NumPy, SciPy și SymPy pentru efectuarea diferitor calcule în limbajul Python.	3
P9. Efectuarea diverselor manipulări asupra unei imagini – schimbarea rezoluției, rotirea, generarea zgomotului, aplicarea filtrelor, segmentarea ș.a. (scipy.ndimage oferă manipularea tablourilor n-dimensionale ca imagini).	4
Total practice:	30

8. Referințe bibliografice

Principale	<ol style="list-style-type: none"> McKinney, Wes. (2017). "Python for Data Analysis." O'Reilly Media. Daniel Chen (2022). "Pandas for Everyone. Python Data Analysis", Pearson Education.
Suplimentare	<ol style="list-style-type: none"> Pedregosa, Fabian, et al. (2011). "Scikit-learn: Machine learning in Python." Journal of Machine Learning Research, 12(Oct), 2825-2830. Python Software Foundation. (Python Documentation). [https://docs.python.org/] Grinberg, Miguel. (2018). "Flask Web Development." O'Reilly Media.

9. Utilizarea IA generativă

Permisivitatea de utilizare	<p>Utilizarea IA generative în cadrul temelor și proiectelor este permisă, cu condiția ca studenții să respecte următoarele reguli:</p> <ul style="list-style-type: none"> IA generativă poate fi utilizată pentru generarea de idei, structuri de text sau cod, dar toate materialele generate trebuie să fie revizuite și ajustate de către student pentru a se asigura că acestea corespund cerințelor academice. Orice utilizare a IA generative trebuie să fie declarată în secțiunea de apendice a fiecărei lucrări, folosind fraza: "În timpul pregătirii acestei lucrări, autorul a utilizat [NUME INSTRUMENT / SERVICIU] în scopul [MOTIV]. După utilizarea acestui instrument/serviciu, autorul a revizuit și editat conținutul după cum a fost necesar și își asumă întreaga responsabilitate pentru conținutul lucrării."
Restricții de utilizare	<p>Studenții nu trebuie să considere IA generativă ca o sursă de încredere pentru informații, deoarece nu oferă referințe clare sau surse documentate.</p> <ul style="list-style-type: none"> Nu este permisă citarea directă a conținutului generat de IA în lucrările academice ca și cum ar fi sursă primară. Activitățile în care este interzis utilizarea IA generativă sunt specificare de profesor și sunt de regulă evaluări intermediare și finale sau care nu presupun activități de dezvoltare a competențelor profesionale.

10. Evaluare

Periodică		Curentă	Lucru individual	Examen
EP 1	EP 2			
Învățământ cu frecvență				
15%	15%	15%	15%	40%
Standard minim de performanță. Prezența și activitatea la prelegeri și lucrări practice. Prezentarea proiectului de an. Obținerea notei minime de „5” la fiecare lucrări practice și proiectul de an.				

11. Criterii de evaluare

Activitate	Componente evaluare	Metodă de evaluare, Criterii de evaluare	Pondere în nota finală a activității	Ponderea în evaluarea disciplinei
Învățământ cu frecvență				
Evaluare periodică I	Conținut teoretic, teme 1-5	Test pe MOODLE	100%	10%
Evaluare periodică II	Conținut teoretic, teme 6-10	Test pe MOODLE	100%	10%
Evaluare curentă	Activitatea practică	Discuții în cadrul orelor de practică	50%	10%
		Raport pentru fiecare lucrare de practică încărcat pe MOODLE	50%	
Lucrul individual/Proiect de an	Cercetare la temă	Referat/Prezentare/discurs public. Raportul încărcat pe MOODLE	100%	30%
Evaluarea finală	Conținut teoretic și practic	Test pe MOODLE	100%	40%