

FIȘA DISCIPLINEI

Anul universitar 2025 - 2026

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
1.2. Facultatea	Științe
1.3. Departament	Departamentul de Matematică și Informatică
1.4. Domeniul de studiu	Informatică
1.5. Ciclul de studii ¹	Licență
1.6. Specializarea	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei		Tehnici avansate de programare		Cod	FSTI.MAI.INF.L.SO.2.2020.E-5.3	
2.2. Titular activități de curs		Lector. Univ. Dr. Maria Flori				
2.3. Titular activități practice		Lector. Univ. Dr. Maria Flori				
2.4. An de studiu ²	1	2.5. Semestrul ³		3	2.6. Tipul de evaluare ⁴	E
2.7. Regimul disciplinei ⁵			O	2.8. Categoria formativă a disciplinei ⁶		S

3. Timpul total estimat

3.1. Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână					
3.1.a.Curs	3.1.b. Seminar	3.1.c. Laborator	3.1.d. Proiect	3.1.e Alte	Total
2		2			4
3.2. Extinderea disciplinei în planul de învățământ – Total ore din planul de învățământ					
3.2.a.Curs	3.2.b. Seminar	3.2.c. Laborator	3.2.d. Proiect	3.2.e Alte	Total ⁷
28		28			56
Distribuția fondului de timp pentru studiu individual ⁸					Nr. ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					16
Tutoriat ⁹					14
Examinări ¹⁰					4
3.3. Total ore alocate studiului individual ¹¹ (NOSIsem)					69
3.4. Total ore din Planul de învățământ (NOADsem)					56
3.5. Total ore pe semestru ¹² (NOADsem + NOSIsem)					125
3.6. Nr ore / ECTS					25
3.7. Număr de credite ¹³					5

4. Precondiții (acolo unde este cazul)

4.1. Discipline necesar a fi promovate anterior (de curriculum) ^{xiv}	-
4.2. Competențe	-

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului ^{xv}	Sală de curs, dotată cu tablă, calculator, videoproiector și software specific.	
5.2. De desfășurare a activităților practice (lab/sem/pr/aplic) ^{xvi}	Sală de laborator, dotată cu tablă, calculatoare, videoproiector și software specific.	

6. Rezultate ale învățării^{xvii}

Numărul de credite alocat disciplinei: 5				
Rezultatele învățării				Repartizare credite pe rezultatele învățării
Nr. crt.	Cunoștințe	Aptitudini	Responsabilitate și autonomie	
RÎ 1	Studentul identifică, explică și argumentează concepte fundamentale de tehnici de programare.	Studentul elaborează, dezvoltă și demonstrează soluții software complexe utilizând tehnici avansate de programare.	Studentul coordonează echipe tehnice pentru dezvoltarea de aplicații informatice, asumând decizii responsabile legate de optimizarea și integrarea acestora.	2
RÎ 2	Studentul identifică, explică și argumentează concepte fundamentale de algoritmi și tehnici de programare.	Studentul elaborează, dezvoltă și demonstrează soluții software complexe utilizând algoritmi și tehnici de programare.	Studentul coordonează echipe tehnice pentru dezvoltarea de aplicații informatice, asumând decizii responsabile legate de optimizarea și integrarea acestora.	2
RÎ 3	Studentul identifică, explică și argumentează paradigme de programare.	Studentul demonstrează soluții software complexe utilizând eficient paradigme diverse de programare.	Studentul coordonează echipe tehnice pentru dezvoltarea de aplicații informatice, asumând decizii responsabile legate de optimizarea și integrarea acestora.	1

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

7.1. Obiectivul general	Formarea unei gândiri algoritmice, capabile să selecteze tehnica optimă de utilizat pentru rezolvarea unei anumite probleme. Cunoașterea tehnicilor de programare: divide et impera, backtracking, greedy, programarea dinamică și branch and bound, precum și caracteristicile pe care trebuie să le satisfacă o problemă pentru a putea fi rezolvată cu una dintre aceste tehnici.
7.2. Obiectivele specifice	Construcția și implementarea în C/C++ a unor algoritmi ce utilizează tehnicile de programare backtracking, greedy, programare dinamică, divide et impera, branch and bound.

8. Conținuturi

8.1. Curs ^{xviii}		Metode de predare ^{xix}	Nr. ore
Curs 1	Tehnica divide et impera. Principiul metodei, analiza eficienței. Algoritmi de sortare obținuți prin tehnica divide et impera	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 2	Tehnica divide et impera. Algoritmi de căutare obținuți prin tehnica divide et impera.	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 3	Recapitulare. Alocarea dinamică a memoriei. Structura de stivă și coadă. Structura de arbore binar de căutare. Tehnica divide et impera aplicată pe structuri dinamice de date. Calculul timpului	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 4	Tehnica backtracking. Principiul metodei, analiza eficienței. Backtracking iterativ	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 5	Tehnica backtracking. Backtracking recursiv. Backtracking în plan	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 6	Tehnica greedy. Principiul metodei, analiza eficienței. Algoritmi greedy optimali.	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 7	Tehnica greedy. Algoritmi greedy euristici	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 8	Tehnica programării dinamice. Principiul metodei, analiza eficienței	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 9	Tehnica programării dinamice. Metoda înainte, înapoi și mixtă	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 10	Tehnica branch and bound. Principiul metodei, analiza eficienței	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 11	Tehnica branch and bound. Exemple de probleme rezolvate	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 12	Algoritmi genetici. Modele generale. Principiul de funcționare și aplicabilitate	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 13	Compararea tehnicilor de programare. Problema comis voiajorului.	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Curs 14	Proiecte complexe cu tehnici de programare	Conversația frontală; utilizarea videoproiectorului; discuții deschise cu studenții.	2
Total ore curs:			28

8.2. Activități practice

Activități practice (8.2.a. Seminar ^{xx} / 8.2.b. Laborator ^{xxi} / 8.2.c. Proiect ^{xxii})	Metode de predare	Nr. ore
--	-------------------	---------

Probleme ce utilizează algoritmi de sortare obținuți prin tehnica divide et impera	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica divide et impera. Probleme ce utilizează algoritmi de căutare obținuți prin tehnica divide et impera. Alți algoritmi de tip divide et impera	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Recapitulare. Structuri dinamice de date. Stive, cozi, arbori binari de cautare	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica divide et impera aplicata pe structuri dinamice de date. Calculul timpului	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica backtracking rezolvare de probleme diverse folosind backtracking iterativ și backtracking recursiv	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica backtracking în plan și rezolvarea de probleme folosind această tehnică	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica backtracking și rezolvarea de probleme diverse folosind probleme de combinatorică, problema damelor, comis voiajorului, colorarea hărților, plata unei sume de bani, etc	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica greedy. Designul și implementarea de algoritmi greedy optimali	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica greedy. Designul și implementarea de algoritmi greedy euristici	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica programării dinamice. Designul și implementarea de algoritmi pentru rezolvarea unor probleme care satisfac principiul optimalității în forma înainte, înapoi, mixtă	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tehnica branch and bound	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Exemplificare de algoritmi genetici. Calculul operatorilor genetici	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Tratarea obiectuală a tehnicilor de programare. Exemplificări de clase și ierarhizări ale acestora	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Probleme ce utilizează algoritmi de sortare obținuți prin tehnica divide et impera	Prezentarea problemelor la tabla si/ videoproiector, discuții cu studenții.	2
Total ore seminar/laborator		28

9. Bibliografie

9.1. Referințe bibliografice recomandate	Dana Simian, Proiectarea algoritmilor: principii si tehnici, Ed. ULBS, 2016 (material disponibil si pe classroom si in biblioteca digitala ULBS)
9.2. Referințe bibliografice suplimentare	Tony Gaddis. Starting out with C++. From Control Structures Through Objects, Ed. a 6-a, Person Education Inc., 2012
	Frank Carrano, Timoty Henry, Data abstraction and problem solving in C++, Walls and Mirrors, Ed. Pearson, 7th Ed. 2021

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Lector univ. dr. Maria Flori	
Responsabil program de studii	Lector univ. dr. Daniel Hunyadi	
Director Departament	Prof.univ.dr. Mugur Acu	

¹ Licență / Master

² 1-4 pentru licență, 1-2 pentru master

³ 1-8 pentru licență, 1-3 pentru master

⁴ Examen, colocviu sau VP A/R – din planul de învățământ

⁵ Regim disciplină: O=Disciplină obligatorie; A=Disciplină opțională; U=Facultativă

⁶ Categoria formativă: S=Specialitate; F=Fundamentală; C=Complementară; I=Asistată integral; P=Asistată parțial; N=Neasistată

⁷ Este egal cu 14 săptămâni x numărul de ore de la punctul 3.1 (similar pentru 3.2.a.b.c.)

⁸ Liniile de mai jos se referă la studiul individual; totalul se completează la punctul 3.37.

⁹ Între 7 și 14 ore

¹⁰ Între 2 și 6 ore

¹¹ Suma valorilor de pe liniile anterioare, care se referă la studiul individual.

¹² Suma (3.5.) dintre numărul de ore de activitate didactică directă (NOAD) și numărul de ore de studiu individual (NOSI) trebuie să fie egală cu numărul de credite alocate disciplinei (punctul 3.7) x nr. ore pe credit (3.6.). De asemenea, se recomandă ca numărul orelor de studiu individual să fie cel puțin egal cu cel al orelor de pregătire universitară formată din ore didactice

¹³ Numărul de credit se calculează după formula următoare și se rotunjește la valori vecine întregi (fie prin micșorare fie prin majorare)

$$\text{Nr. credite} = \frac{\text{NOCpSpD} \times C_C + \text{NOApSpD} \times C_A}{\text{TOCpSpD} \times C_C + \text{TOApSpD} \times C_A} \times 30 \text{ credite}$$

Unde:

- NOCpSpD = Număr ore curs/săptămână/disciplina pentru care se calculează creditele
- NOApSpD = Număr ore aplicații (sem./lab./pro.)/săptămână/disciplina pentru care se calculează creditele
- TOCpSpD = Număr total ore curs/săptămână din plan
- TOApSpD = Număr total ore aplicații (sem./lab./pro.)/săptămână din plan
- C_C/C_A = Coeficienți curs/aplicații calculate conform tabelului

Coeficienți	Curs	Aplicații (S/L/P)
Licență	2	1
Master	2,5	1,5
Licență lb. străină	2,5	1,25

^{xiv} Se menționează disciplinele obligatorii a fi promovate anterior sau echivalente

^{xv} Tablă, videoproiector, flipchart, materiale didactice specifice, platforme on-line etc.

^{xvi} Tehnică de calcul, pachete software, standuri experimentale, platforme on-line etc.

^{xvii} Rezultatele învățării se vor menționa în funcție de standardele specifice ale Comisiilor de specialitate ARACIS (<https://www.aracis.ro/ghiduri/>)

^{xviii} Titluri de capitole și paragrafe

^{xix} Expunere, prelegere, prezentare la tablă a problematicei studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

^{xx} Discuții, dezbateri, prezentare și/sau analiză de lucrări, rezolvare de exerciții și probleme

^{xxi} Demonstrație practică, exercițiu, experiment

^{xxii} Studiu de caz, demonstrație, exercițiu, analiza erorilor etc.

^{xxiii} Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

^{xxiv} CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală; N/A – nu se aplică

^{xxv} Se va preciza numărul de teste și săptămânile în care vor fi susținute.

^{xxvi} Cercuri științifice, concursuri profesionale etc.

^{xxvii} Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.