

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică din Cluj-Napoca
1.2 Facultatea	Facultatea de Științe (Centrul Universitar Nord din Baia Mare)
1.3 Departamentul	Departamentul de Matematică și Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Licență
1.6 Programul de studii / Calificarea	Informatică (Baia Mare, limba română) / Licențiat în informatică
1.7 Forma de învățământ	IF – învățământ cu frecvență

2. Date despre disciplină

2.1 Denumirea disciplinei	Algoritmi fundamentali și structuri de date II	Codul disciplinei	15.00
2.2 Titularul de curs	Conf. univ. dr. Cosmin Nicolae SABO – cosmin.sabo@mi.utcluj.ro		
2.3 Titularul activităților de seminar / laborator / proiect / practică	Asistent universitar, post vacant		
2.4 Anul de studiu	I	2.5 Semestrul	2
		2.6 Tipul de evaluare	E
2.7 Regimul disciplinei	Categoriza formativă		DF
	Opționalitate		DOB

3. Timpul total estimat

3.1 Număr de ore pe săptămână	4	din care:	3.2 Curs	2	3.3 Seminar	-	3.3 Laborator	2	3.3 Proiect	-	3.3 Practică	-
3.4 Număr de ore pe semestru	56	din care:	3.5 Curs	28	3.6 Seminar	-	3.6 Laborator	28	3.6 Proiect	-	3.3 Practică	-
3.7 Distribuția fondului de timp (ore pe semestru) pentru studiu individual și evaluare:												
(a) Evaluare												6
(b) Studiul după manual, suport de curs, bibliografie și notițe												20
(c) Documentare suplimentară în bibliotecă, pe platforme electronice de specialitate și pe teren												8
(d) Pregătire seminarii / laboratoare, teme, referate, portofolii și eseuri												30
(e) Tutoriat												3
(f) Alte activități												2
3.8 Total ore studiu individual și evaluare (suma (3.7(a))...3.7(f))								69				
3.9 Total ore pe semestru (3.4+3.8)								125				
3.10 Numărul de credite								5				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Algoritmi fundamentali și structuri de date I (recomandată).
4.2 de competențe	Programare de bază în limbajul de programare utilizat: variabile, structuri de control, funcții, liste, dicționare, fișiere; algoritmi fundamentali de căutare și sortare; noțiunea de complexitate. Primele două săptămâni reiau recursivitatea și sortarea, ca punte pentru studenții care au nevoie de nivelare.

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu videoproiector și acces la internet; cursul îmbină aspectele
--------------------------------	--

	teoretice cu exemple și demonstrații, pe care studenții le pot urmări și rula pe laptopul personal. Suportul de curs, exemplele rezolvate, testele de autoevaluare și anunțurile se publică pe platforma Campus Virtual. Cursul este comun programelor Matematică informatică și Informatică.
5.2. de desfășurare a seminarului / laboratorului / proiectului	Laborator cu calculatoare pe care sunt instalate interpretorul limbajului de programare utilizat (Python, versiune curentă), un editor de cod și un sistem de control al versiunilor; se recomandă lucrul pe laptopul personal. Activitatea este individuală la sarcinile de laborator și în echipe de 2-3 studenți la proiect; echipele se constituie în săptămâna 7, iar proiectul are trei etape notate. Livrabilele se predau prin Campus Virtual până la termenele anunțate și sunt verificate automat, cu feedback la laborator; codul proiectului este versionat, cu contribuția fiecărui membru vizibilă în istoric. Asistenții bazați pe inteligență artificială sunt permisi la laborator și la proiect, cu declararea lor în fiecare livrabil și cu obligația de a explica propriul cod; sunt interziși la teste și la examen. Prezența la laborator este obligatorie, conform regulamentului UTCN privind activitatea profesională a studenților.

6. Competențele specifice acumulate

Competențe profesionale	C1. Programarea în limbaje de nivel înalt: implementarea structurilor de date și a algoritmilor studiați într-un limbaj de nivel înalt, inclusiv cu bibliotecile standard. C2. Dezvoltarea și întreținerea aplicațiilor informatice: dezvoltarea în echipă a unei aplicații mici bazate pe structuri de date, cu cod versionat, testat și documentat. C4. Utilizarea bazelor teoretice ale informaticii și a modelelor formale: structuri de date fundamentale, tehnici de proiectare a algoritmilor, analiza complexității, noțiunea de problemă dificilă.
Competențe transversale	CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic-științific, pentru valorificarea creativă a propriului potențial, cu respectarea principiilor și a normelor de etică profesională. CT2. Lucrul în echipă: asumarea de roluri, planificarea și urmărirea sarcinilor, comunicarea în cadrul echipei. CT3. Utilizarea unor metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, de adaptare la cerințele unei societăți dinamice și de comunicare în limba română și într-o limbă de circulație internațională.

7. Rezultatele așteptate ale învățării

Cunoștințe	RI1. Descrie structurile de date fundamentale (liste înlănțuite, stive și cozi, heap și coadă cu prioritate, arbori, arbori binari de căutare, tabele de dispersie, grafuri), operațiile lor și costul fiecărei operații. RI2. Explică tehnicile de proiectare a algoritmilor (divide et impera, Greedy, backtracking, programare dinamică), condițiile în care fiecare este adecvată și exemple în care una eșuează și alta reușește. RI3. Explică analiza complexității algoritmilor recursivi (arbori de recursie), compară algoritmi de sortare și de căutare studiați și cunoaște, la nivel introductiv, noțiunea de problemă dificilă (clasele P și NP).
------------	---

Abilități	RI4. Implementează structurile de date studiate și le folosește, inclusiv în variantele din biblioteca standard a limbajului, pentru a rezolva probleme date. RI5. Aplică tehnicile de proiectare a algoritmilor la probleme noi, formulează subproblemele și argumentează alegerea și corectitudinea soluției pe exemple și contraexemple. RI6. Modelează probleme reale cu grafuri și arbori și aplică parcurgeri, drumuri minime și arbori parțiali de cost minim. RI7. Analizează și măsoară experimental complexitatea soluțiilor și alege structura de date potrivită după operațiile dominante.
Responsabilitate și autonomie	RI8. Realizează în echipă un proiect software mic, cu rol asumat, planificare pe etape, cod versionat și contribuție individuală vizibilă, pe care îl prezintă și îl susține în fața colegilor. RI9. Își verifică soluțiile pe cazuri de test, inclusiv cazuri limită, documentează codul și declară sursele și instrumentele de asistență folosite, fiind capabil să explice fiecare parte a codului propriu.

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

8.1 Obiectivul general al disciplinei	Formarea capacității de a alege și a implementa structura de date și tehnica de proiectare a algoritmilor potrivite unei probleme date, cu înțelegerea costurilor, prin exemple rezolvate, exersare săptămânală și un proiect de echipă.
8.2 Obiectivele specifice	O1. Aprofundarea recursivității și a paradigmei divide et impera pe algoritmi de sortare eficienți. O2. Implementarea și utilizarea structurilor de date liniare și neliniare: liste înlănțuite, heap, arbori, arbori binari de căutare, tabele de dispersie, grafuri. O3. Modelarea problemelor reale cu grafuri și aplicarea parcurgerilor și a algoritmilor de drum minim și de arbore parțial de cost minim. O4. Aplicarea tehnicilor Greedy, backtracking și programare dinamică, cu înțelegerea limitelor fiecăreia. O5. Analiza complexității algoritmilor recursivi și alegerea structurii de date după costul operațiilor. O6. Dezvoltarea în echipă a unui proiect software mic, cu cod versionat, testat, documentat și prezentat.

9. Conținuturi

9.1 Curs	Nr. ore	Metode de predare	Observații
1. Recursivitate și divide et impera: sortarea prin interclasare; arbori de recursie; complexitate.	2	Prelegere interactivă cu aspectele teoretice și exemple; fiecare structură se vede întâi în varianta încorporată a limbajului, apoi se implementează simplificat; suport de curs, exemple și teste de autoevaluare pe Campus Virtual.	Tema cursului continuă la laborator. Teste scurte în săpt. 5 și 10. Se cere nucleul din suport.
2. Sortarea rapidă și selecția; alegerea pivotului, randomizare; sortarea în practică.	2		
3. Liste înlănțuite simple și duble; stive și cozi pe liste înlănțuite; comparație cu lista încorporată.	2		
4. Heap și coada cu prioritate; sortarea cu heap.	2		
5. Arbori: reprezentare, parcurgeri, arbori de expresii.	2		
6. Arbori binari de căutare: căutare, inserare, ștergere; ideea de echilibrare.	2		
7. Tabele de dispersie: funcții hash, coliziuni; dicționarele și mulțimile încorporate.	2		
8. Grafuri (I): reprezentări; parcurgeri în lățime și în adâncime; componente conexe; drum minim în graf neponderat.	2		
9. Grafuri (II): drumuri minime în grafuri ponderate; arbori parțiali de cost minim.	2		
10. Tehnica Greedy: alegerea local optimă; exemple și contraexemple; coduri Huffman.	2		
11. Backtracking: schema generală, arborele stărilor, tăierea ramurilor.	2		
12. Programare dinamică (I): subprobleme suprapuse, memoizare, tabulare, reconstruirea soluției.	2		
13. Programare dinamică (II): probleme pe secvențe; distanța de editare, cel mai lung subșir comun.	2		
14. Sinteză: alegerea structurii de date și a tehnicii pentru o problemă dată; probleme dificile (P și NP, introductiv); pregătirea examenului.	2		
Bibliografie			
1. Suportul de curs al disciplinei, publicat pe Campus Virtual.			
2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., Introduction to Algorithms, ed. a 4-a, MIT Press, 2022.			
3. Miller, B. N., Ranum, D. L., Problem Solving with Algorithms and Data Structures using Python, ed. a 3-a, Runestone Academy; ediție interactivă gratuită, https://runestone.academy/ns/books/published/pythonds3/ .			
4. Goodrich, M. T., Tamassia, R., Goldwasser, M. H., Data Structures and Algorithms in Python, Wiley, 2013.			
5. Sedgewick, R., Wayne, K., Algorithms, ed. a 4-a, Addison-Wesley, 2011; materiale și vizualizări gratuite, https://algs4.cs.princeton.edu .			
6. Skiena, S. S., The Algorithm Design Manual, ed. a 3-a, Springer, 2020.			
7. Cormen, T. H., Algorithms Unlocked, MIT Press, 2013 (introducere accesibilă).			
8. Giumale, C., Introducere în analiza algoritmilor: teorie și aplicație, Polirom, 2004.			
9. Halim, S., Halim, F., VisuAlgo, https://visualgo.net ; culegeri de probleme cu evaluare automată în limba română: https://infoarena.ro , https://www.pbinfo.ro .			

9.2 Seminar / laborator / proiect / practică	Nr. ore	Metode de predare	Observații
1. Sortarea prin interclasare și numărarea comparațiilor (L1).	2	Lucru individual pe calculator, pornind de la problema săptămânii începută la curs, și lucru în echipe de 2-3 la proiect; sarcini pe două niveluri (de bază și extindere), verificate automat, cu feedback la laborator; evaluare între colegi la raportul intermediar; fișe de laborator și soluții de referință pentru titularul de laborator; asistenți AI permiși cu declarare; predare prin Campus Virtual.	Pe subgrupe. Lucrări notate L1-L7 (săpt. impare). Proiect de echipă în 3 etape (săpt. 8, 12, 14), pe teme date, cu cod versionat.
2. Compararea experimentală a algoritmilor de sortare; al k-lea cel mai mic element.	2		
3. Listă înlănțuită cu operațiile de bază; anulare și refacere cu stive (L2).	2		
4. Planificator de sarcini cu priorități pe heap.	2		
5. Arbore de expresii: construire, parcurgeri, evaluare (L3); test scurt T1 (săpt. 1-4).	2		
6. Index de căutare într-un text cu arbore binar de căutare.	2		
7. Tabelă de dispersie proprie și comparație cu dicționarul încorporat (L4); constituirea echipelor de proiect.	2		
8. Rezolvator de labirint prin parcurgeri de graf; propunerea proiectului de echipă (P1).	2		
9. Navigație pe o hartă dată: drumuri minime (L5).	2		
10. Compresor de text cu coduri Huffman; test scurt T2 (săpt. 5-9).	2		
11. Rezolvator de puzzle-uri prin backtracking (L6).	2		
12. Optimizator de resurse cu constrângeri prin programare dinamică; raport intermediar de proiect cu evaluare între colegi (P2).	2		
13. Comparator de fișiere pe baza celui mai lung subșir comun (L7).	2		
14. Prezentarea și susținerea proiectelor de echipă (P3).	2		
Bibliografie			
1. Îndrumarul de laborator al disciplinei (fișe de laborator, șabloane de cod, teste automate, temele și rubrica proiectului), publicat pe Campus Virtual.			
2. Python Software Foundation, The Python Standard Library (modulele pentru colecții, cozi cu prioritate, măsurarea timpului), https://docs.python.org/3/library/ .			
3. Miller, B. N., Ranum, D. L., Problem Solving with Algorithms and Data Structures using Python, ediția interactivă, https://runestone.academy/ns/books/published/python3/ .			
4. Halim, S., Halim, F., VisuAlgo, https://visualgo.net .			
5. Chacon, S., Straub, B., Pro Git, ed. a 2-a, Apress, 2014, disponibilă gratuit la https://git-scm.com/book (cap. 1-3).			
6. Culegeri de probleme cu evaluare automată: https://infoarena.ro , https://www.pbinfo.ro .			

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Structurile de date și tehnicile de proiectare a algoritmilor sunt conținutul verificat cel mai frecvent la interviurile tehnice pentru posturile de programator și dezvoltator software, în firmele din regiune și în cele internaționale, și fac parte din nucleul obligatoriu al recomandărilor curriculare internaționale

(ACM/IEEE-CS Computer Science Curricula 2023, ariile „Algorithmic Foundations” și „Software Development Fundamentals”) și al standardelor ARACIS pentru domeniile Informatică și Matematică. Proiectul de echipă, cu cod versionat și prezentat, dă absolvenților un livrabil pe care angajatorii îl pot verifica direct și corespunde ocupațiilor din ESCO/COR vizate de programe: programator de sistem informatic (251204) pentru Informatică, matematician și consilier statistician (212009, 212011) pentru Matematică informatică.

Disciplina continuă Algoritmi fundamentali și structuri de date I și pregătește Programarea orientată pe obiecte, Baze de date, Tehnici avansate de programare, Inteligență artificială și proiectele de licență; ordinea temelor și abordarea prin probleme rezolvate urmează cursurile de referință de structuri de date din comunitatea academică.

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare (și forma evaluare: continuă/sumativă)	11.3 Pondere din nota finală
11.4 Curs	Cunoașterea și înțelegerea conceptelor (RI1-RI3): structurile de date și costul operațiilor, tehnicile de proiectare a algoritmilor și limitele lor, analiza complexității. Capacitatea de a rezolva individual, pe calculator, probleme noi alegând structura de date și tehnica potrivite, cu cod corect, lizibil și testat (RI4-RI7).	Evaluare continuă: două teste scurte pe Campus Virtual (T1 în săpt. 5, materia săpt. 1-4; T2 în săpt. 10, materia săpt. 5-9), corectate automat (10%). Evaluare sumativă, examen în sesiune: probă practică individuală pe calculator, 2 ore, trei probleme de dificultate graduală, fără instrumente de asistență, verificată automat și citită de titular (30%); chestionar conceptual pe Campus Virtual, 30 de minute (10%).	50% (10% teste scurte, 30% probă practică, 10% chestionar conceptual)

11.5 Seminar/Laborator /Proiect / practică	Portofoliul de laborator (RI4-RI7, RI9): sarcinile săptămânale cu nivelul de bază trecut de verificarea automată și lucrările L1-L7, cu cod corect, lizibil și testat; respectarea termenelor; declararea instrumentelor de asistență. Proiectul de echipă (RI4-RI6, RI8, RI9): alegerea și implementarea corectă a structurilor de date și a algoritmilor, calitatea codului și a testelor, documentația, prezentarea și susținerea, contribuția individuală vizibilă în istoricul codului.	Evaluare continuă: portofoliul de laborator, predat prin Campus Virtual, verificat automat și notat pe rubrică de titularul de laborator (25%). Evaluare continuă: proiectul de echipă (2-3 studenți), pe teme date, în trei etape notate pe rubrică: P1 propunere (săpt. 8, 20%), P2 raport intermediar cu evaluare între colegi (săpt. 12, 30%), P3 livrare și prezentare (săpt. 14, 50%); nota individuală poate diferi de a echipei cu până la 2 puncte, după contribuția vizibilă în istoricul codului și după susținere (25%).	50% (25% portofoliul de laborator, 25% proiectul de echipă)
--	--	--	--

11.6 Standard minim de performanță

Nota finală $N = 0,10 \times TS + 0,25 \times PL + 0,25 \times PR + 0,30 \times EP + 0,10 \times EC$, unde TS = media testelor scurte, PL = portofoliul de laborator, PR = proiectul de echipă (cu ajustarea individuală), EP = proba practică, EC = chestionarul conceptual, toate notate de la 1 la 10. Evaluarea continuă (TS, PL, PR) reprezintă 60% din nota finală, evaluarea sumativă (EP, EC) 40%.

Condiții minime pentru nota 5: toate cele șapte lucrări L1-L7 predate și $PL \geq 5$; proiectul livrat integral (P1-P3), funcțional în cerințele de bază, cu contribuție individuală vizibilă în istoricul codului și $PR \geq 5$; ambele teste scurte susținute; la proba practică, studentul rezolvă singur, complet și corect, cel puțin o problemă și obține $EP \geq 5$; nota finală $N \geq 5$.

Livrabilele întârziate pierd un punct pe săptămână din nota livrabilului. În sesiunea de restanțe se reface numai examenul (proba practică și chestionarul conceptual); testele, portofoliul și proiectul rămân valabile în anul universitar curent. Instrumentele de asistență se declară în fiecare livrabil și sunt interzise la teste și la examen; nedeclararea sau incapacitatea de a explica propriul cod se tratează ca abatere de la integritatea academică, conform regulamentelor UTCN.

Data completării:	Titulari	grad didactic, titlu Prenume NUME	Semnătura
10.09.2026	Curs	Conf. univ. dr. Cosmin Nicolae SABO	
	Aplicații	Asistent universitar, post vacant	
		-	

Avizul Directorului departamentului care coordonează disciplina, doar dacă acesta diferă de departamentul organizator al programului de studii	Nu este cazul (disciplina este coordonată de departamentul organizator al programului de studii).
--	---

Data avizării în Consiliul Departamentului: 14.09.2026

Director Departament DMI

Conf. univ. dr. Cosmin Nicolae SABO

Data aprobării în Consiliul Facultății: 17.09.2026

Decan,

Conf. univ. dr. Monica Liliana MARIAN