



FIȘA DISCIPLINEI
ANUL UNIVERSITAR 2025 – 2026

1. Date despre program

1.1. Instituția de învățământ superior	UNIVERSITATEA DIN CRAIOVA
1.2. Facultatea	Automatică, Calculatoare și Electronică
1.3. Departamentul	Calculatoare și Tehnologia Informației
1.4. Domeniul de studii	Calculatoare și Tehnologia Informației
1.5. Ciclu de studii universitare	Licență
1.6. Forma de organizare	IF
1.7. Programul de studii	Calculatoare / L2060101010

2. Date despre disciplină

2.1. Denumirea disciplinei	Proiectarea algoritmilor						
2.2. Titularul activităților de curs	Bădică Costin						
2.3. Titularul activităților de seminar/ laborator							
2.4. Anul de studiu	1	2.5. Semestrul	2	2.6. Tipul de evaluare	E	2.7. Regimul disciplinei	DOB

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

3.1. Numărul de ore pe săptămână	5	din care: 3.2 curs	2	3.3. sem./lab./pr.	1/2
3.4. Total ore din planul de învățământ	70	din care: 3.5 curs	28	3.6. sem./lab./pr.	14/28
Distribuția fondului de timp - ore/sapt.					
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					15
Tutoriat					5
Examinări					5
Alte activități.....					
3.7. Total ore studiu individual					55
3.8. Total ore pe semestru					125
3.9. Numărul de credite					5

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Programarea calculatoarelor și limbaje de programare Analiză matematică
4.2. de competențe	n/a

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Cursul este predat utilizând prelegeri interactive față-în-față (f2f) și/sau online, distribuirea materialelor de curs în format digital și interacțiunea cu studenții. Procesul de predare este structurat astfel: - 70% prezentări teoretice, folosind suportul de curs (slide-uri), precum și exemple de exerciții teoretice și demonstrații și tutoriale practice.
--------------------------------	---

	- 30% interacțiune (discuții cu studenții) bazată pe întrebări, exerciții și probleme propuse.
5.2. de desfășurare a seminarului/ laboratorului	Seminarul presupune rezolvarea de exerciții și probleme algoritmice folosind tehnicile și metodele prezentate la prelegeri. Laboratorul impune realizarea de experimente de algoritmi utilizând instrumente de programare și dezvoltare adecvate domeniului. Laboratorul urmărește dezvoltarea competențelor studenților pentru: - Implementarea metodelor de proiectare, implementare a algoritmilor. - Testarea programelor algoritmilor în diverse scenarii experimentale.

6. Obiectivele disciplinei - rezultate așteptate ale învățării la formarea cărora contribuie parcurgerea și promovarea disciplinei

Cunoștințe	Studentul/Absolventul: 1. Identifică și descrie concepte, principii și metode de bază din informatică. 2. Explică și interpretează rezultate teoretice și experimentale din informatică. 3. Descrie, identifică, rezumă concepte și metode elementare privitoare la limbaje de programare, medii de programare, tehnici de programare și modul lor de aplicare în probleme concrete. 1. Va cunoaște algoritmi fundamentali, structuri de date, teoria grafurilor și programarea dinamică pentru a implementa eficient soluții algoritmice la probleme bine definite.
Aptitudini (Abilități)	Studentul/Absolventul: 4. Operează cu concepte, principii și metode de bază din informatică. 5. Aplică criterii și metode de evaluare pentru identificarea, proiectarea, experimentarea, analiza și aprecierea calitativă și cantitativă a algoritmilor. 6. Alege și explică concepte proprii specifice proiectării algoritmilor. 7. Va avea aptitudini în ceea ce privește stăpânirea algoritmilor și structurilor de date fundamentale, codarea eficientă (rapidă și corectă cu constrângeri în ceea ce privește timpul de rulare și memoria utilizată) de soluții la probleme bine definite și de lucru în echipă.
Responsabilitate și autonomie	Studentul/Absolventul: 1. Practică raționamentul logic, evaluarea și auto-evaluarea în luarea deciziilor. 2. Aplică cele mai bune practici în proiectarea și implementarea algoritmilor, incluzând scrierea de cod eficient și bine documentat. 3. Manifestă un comportament onorabil, responsabil și etic, în spiritul legii, pentru a asigura reputația profesiei.

7. Conținuturi

7.1. CURS	Modalitatea de desfășurare	Metode de predare	Fond de timp alocat (ore)
1. Introducere în algoritmi. Analiza complexității algoritmilor. Notăția asimptotică.	față în față	Prelegeri. Exemple teoretice. Demonstrații practice. Interacțiune prin întrebări și răspunsuri.	2
2. Analiza algoritmilor folosind recurențe. Teorema "Master".			2
3. Corectitudinea și testarea algoritmilor, Inducție și invarianți.			2
5. Algoritmi „Divide et impera”. Sortarea prin			2

interclasare. Problema broker-ului de acțiuni. Înfășurătoarea convexă.			
25. Sortarea rapidă. Algoritmi cu randomizare. Selecția. Sortarea prin numărare. Universul algoritmilor de sortare. Limita inferioară a sortării prin comparație.			2
6. Tipuri de date abstracte. Liste. Programare modulară.			2
7. Stive și cozi. Alocarea dinamică a memoriei.			2
8. Grafuri și arbori: elemente de bază. Reprezentarea și parcurgerea arborilor.			2
9. Reprezentarea și parcurgerea grafurilor. Parcurgerea în adâncime și parcurgerea pe niveluri. Sortarea topologică.			2
10. Programare dinamică. Înmulțirea optimă a unui șir de matrice. Problema celei mai lungi subșiruri comune. Problema domino.			2
11. Căutarea exhaustivă și backtracking. Algoritmi combinatoriali. Problema rucsacului - versiunea discretă. Problema reginelor.			2
12. Algoritmi Greedy. Programarea activităților (sau Planificarea activităților). Problema rucsacului - versiunea continuă. Colorarea grafurilor.			2
13. Introducere în NP-completitudine. Clase de probleme. Probleme NP-dure (NP-hard) și NP-complete. Reducerea polinomială. Probleme clasice NP-complete.			2
14. Evaluare și laboratoare restante			2
Bibliografie:			
1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Cliff Stein, Introduction to Algorithms (4th edition), MIT Press and McGraw-Hill, 2022 2. Steven S. Skiena, The Algorithm Design Manual (2nd ed.), Springer, 2011 3. Antti Laaksonen, Guide to Competitive Programming. Learning and Improving Algorithms Through Contests. Springer, 2017 4. Steven S. Skiena, Miguel A. Revilla, Programming Challenges: The Programming Contest Training Manual, Springer, 2003			

7.2. Seminar/laborator/proiect	Modalitatea de desfășurare	Metode de predare	Fond de timp alocat (ore)
---------------------------------------	----------------------------	-------------------	---------------------------

Seminar			
1. Exerciții care acoperă corectitudinea algoritmilor și analiza complexității algoritmilor.	față în față	Tutoriale. Probleme rezolvate. Interacțiune prin întrebări și răspunsuri.	2
2. Exerciții cu algoritmi de tip Divide et Impera.			2
3. Exerciții cu algoritmi pentru liste, stive și cozi.			2
4. Exerciții cu algoritmi pe grafuri și arbori.			2
5. Exerciții cu algoritmi de programare dinamică.			2
6. Exerciții cu algoritmi combinatoriali și backtracking.			2
7. Exerciții cu algoritmi Greedy.			2
Laborator			
1. Introducere în procedura de lucru a laboratorului. Experimente cu sortarea prin inserție.	față în față	Tutoriale. Demonstrații practice și experiențe practice (hands-on). Interacțiune bazată pe întrebări și răspunsuri. Metode, medii și experiențe practice de programare competitivă.	2
2. Căutare binară. Interclasare (sau Îmbinare). Sortare prin interclasare. Sortare prin selecție.			2
3. Testarea algoritmilor. Implementarea algoritmilor în medii de programare competitivă.			2
4. Implementarea algoritmilor de tip Divide et Impera.			2
5. Sortare rapidă și selecție. Algoritmi randomizați.			2
6. Implementarea listelor. Algoritmi care utilizează stive și cozi.			2
7. Reprezentarea și parcurgerea arborilor.			2
8. Algoritmi cu grafuri.			2
9. Algoritmi de programare dinamică.			2
10. Algoritmi combinatoriali.			2
11. Algoritmi <i>backtracking</i> .			2
12. Algoritmi Greedy.			2
13. Evaluare și recuperare.			2
14. Evaluare și recuperare.			2
Bibliografie:			
1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Cliff Stein, Introduction to Algorithms (4th edition), MIT Press and McGraw-Hill, 2022			
2. Steven S. Skiena, The Algorithm Design Manual (2nd ed.), Springer, 2011			
3. Antti Laaksonen, Guide to Competitive Programming. Learning and Improving Algorithms Through Contests. Springer, 2017			
4. Steven S. Skiena, Miguel A. Revilla, Programming Challenges: The Programming Contest Training Manual, Springer, 2003			

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

Conținutul prelegerilor și al aplicațiilor se aliniază cu cel al cursurilor similare de la alte universități naționale și internaționale. Rezultatele învățării și competențele dobândite corespund cerințelor marilor angajatori. Acestea au fost dezvoltate și îmbunătățite continuu pe baza informațiilor primite de la reprezentanții companiilor IT locale. Conținutul prelegerilor și al aplicațiilor se aliniază cu cel al cursurilor similare de la alte universități naționale și internaționale. Rezultatele învățării și competențele dobândite corespund cerințelor marilor angajatori. Acestea au fost dezvoltate și îmbunătățite continuu pe baza informațiilor primite de la reprezentanții companiilor IT locale.

9. Evaluare

Tip activitate	9.1. Criterii de evaluare	9.2. Metode de evaluare	9.3. Pondere din nota finală
9.4. Curs	Rezolvarea de exerciții și probleme care implică metode și tehnici de proiectare și analiză a algoritmilor.	Examen scris.	70 %
9.5. Seminar/laborator/proiect	Rezultate practice obținute în timpul sesiunilor de laborator. Experiență practică în proiectarea și analiza algoritmilor în timpul sesiunilor de seminar.	Cunoștințele practice acumulate sunt evaluate continuu în timpul sesiunilor de laborator și seminar, prin aprecierea rezultatelor practice și a răspunsurilor la întrebări.	30 %
9.6. Standard minim de performanță			
Nota minimă 5.			

Data completării
01.10.2025

Titular de disciplină,
Prof.dr.ing. Costin BĂDICĂ

Semnătura titularului

.....

Data avizării în departament
03.10.2025

Director de departament,
Ș.l.dr.ing. Nicolae ENESCU

Semnătura directorului de departament,

.....

Decan,
Prof.dr.ing. Dorin ȘENDRESCU

Semnătura decanului

.....