

## FIȘA DISCIPLINEI

### 1. Date despre program

|                     |                                                                              |
|---------------------|------------------------------------------------------------------------------|
| Facultatea          | Facultatea de Inginerie Electrică și Știința Calculatoarelor                 |
| Departamentul       | Departamentul de Calculatoare                                                |
| Domeniul de studii  | Calculatoare și tehnologia informației                                       |
| Ciclul de studii    | Program postuniversitar                                                      |
| Programul de studii | Program de conversie profesională – Tehnologia informației și comunicațiilor |

### 2. Date despre disciplină

|                       |                                                                                                    |           |   |                   |     |
|-----------------------|----------------------------------------------------------------------------------------------------|-----------|---|-------------------|-----|
| Denumirea disciplinei | <b>PROIECTAREA ALGORITMIILOR</b>                                                                   |           |   |                   |     |
| Anul de studiu        | II                                                                                                 | Semestrul | 4 | Tipul de evaluare | E   |
| Regimul disciplinei   | Categorია formativă a disciplinei<br>DF - fundamentală, DS - de specializare, DC – complementară   |           |   |                   | DS  |
|                       | Categorია de opționalitate a disciplinei:<br>DOB – obligatorie, DOP – opțională, DFA - facultativă |           |   |                   | DOB |

### 3. Timpul total estimat (ore alocate activităților didactice)

|                                                          |    |      |    |         |   |                            |    |         |   |
|----------------------------------------------------------|----|------|----|---------|---|----------------------------|----|---------|---|
| I a) Număr de ore pe săptămână                           | 4  | Curs | 2  | Seminar | 0 | Laborator/lucrări practice | 2  | Proiect | 0 |
| I b) Totalul de ore pe semestru din planul de învățământ | 56 | Curs | 28 | Seminar | 0 | Laborator/lucrări practice | 28 | Proiect | 0 |

|                                          |     |
|------------------------------------------|-----|
| Distribuția fondului de timp pe semestru | ore |
| II.a) Studiu individual                  | 116 |
| II.b) Tutoriat (pentru ID)               |     |
| III. Examinări                           | 3   |
| IV. Alte activități (precizați):         |     |

|                                              |     |
|----------------------------------------------|-----|
| Total ore studiu individual (II.a+II.b+III)  | 116 |
| Total ore pe semestru (I.b+II.a+II.b+III+IV) | 175 |
| Numărul de credite                           | 7   |

### 4. Competențe specifice acumulate

|                                  |                                                                                                                                                                                                                                                                                 |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competențe profesionale/generale | CP1. Operarea cu fundamente științifice, ingineresti și ale informaticii<br>CP2. Proiectarea componentelor hardware, software și de comunicații<br>CP3. Îmbunătățirea performanțelor sistemelor hardware, software și de comunicații<br>CP6. Proiectarea sistemelor inteligente |
| Competențe transversale          | -                                                                                                                                                                                                                                                                               |

### 5. Rezultatele învățării

| Cunoștințe                                                                                                                                                                                                                                                                      | Aptitudini                                                                                                                                                                                                              | Responsabilitate și autonomie                                                                                                  |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| Studentul/absolventul descrie, identifică, sumarizează concepte și metode elementare privitoare la limbaje de programare, medii de programare, tehnici de programare, baze de date, inteligență artificială și inginerie software și modul lor de aplicare în probleme concrete | Studentul/absolventul alege și explică concepte proprii specifice proiectării algoritmilor.<br>Studentul/absolventul specifică cerințe, analizează, elaborează, dezvoltă și testează programe în limbaje de programare. | Studentul/absolventul are o comportare onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei. |

### 6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

|                                   |                                                                                                                                                                                                                                                                 |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Obiectivul general al disciplinei | Introducere în proiectarea și analiza algoritmilor prin prezentarea, exemplificarea, analiza și discutarea diverselor metode de proiectare a algoritmilor, evaluare a complexității teoretice și implementare de soluții algoritmice pentru probleme specifice. |
|-----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## 7. Conținutul predării și învățării

| Curs                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Nr. ore | Metode de predare                                                | Observații |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|------------------------------------------------------------------|------------|
| <b>1. Modele de calcul și analiză a complexității algoritmilor</b><br>1.1. Instrumente de măsurare a complexității unui algoritm. Complexitatea temporală. Complexitatea spațială. Ordine de complexitate. Notățiile asimptotice $O$ , $\Theta$ și $\Omega$ . Exemple.<br>1.2. Demonstrarea modului de analiză a complexității algoritmilor pentru probleme specifice.                                                                                                                                                                                                                                                                                                                | 3       | Expunerea, prelegerea, conversația, exemplificarea, demonstrația |            |
| <b>2. Metoda Divide et Impera: introducere</b><br>2.1. Prezentarea principiilor metodei Divide et Impera.<br>2.2. Introducerea metodei Divide et Impera prin discutarea și analiza unor probleme exemplu (determinarea valorii extreme dintr-o mulțime de numere reale, algoritmul căutării binare) și realizarea de comparații cu alte soluții algoritmice pentru rezolvarea acestor probleme.<br>2.3. Analiza complexității algoritmilor Divide et Impera. Teorema Master. Exemple de aplicare a teoremei Master pentru diverse probleme.                                                                                                                                           | 3       |                                                                  |            |
| <b>3. Metoda Divide et Impera: aplicații</b><br>3.1. Algoritmul sortării prin interclasare.<br>3.2. Algoritmul sortării rapide.<br>3.3. Analiza comparativă a performanțelor diverselor metode de sortare a datelor. Realizarea de conexiuni cu alți algoritmi studiați în cadrul unor discipline anterioare. Identificarea algoritmilor de sortare implementați în cadrul diverselor platforme de dezvoltare a aplicațiilor software.<br>3.4. Demonstrarea aplicării metodei Divide et Impera pentru diverse probleme specifice, e.g., problema determinării numărului de inversiuni ale unei permutări, problema tăieturilor, probleme de partiționare a figurilor geometrice, etc. | 3       |                                                                  |            |
| <b>4. Metoda programării dinamice: introducere</b><br>4.1. Prezentarea principiilor metodei programării dinamice. Rezolvarea problemelor algoritmice folosind principiile programării dinamice.<br>4.2. Exemplificarea metodei prin analiza unor probleme specifice, e.g., șirul lui Fibonacci, calculul combinărilor $C_{n,k}$ . Realizarea de comparații cu alte abordări pentru rezolvarea acestor probleme și analiza complexității diverselor abordări.                                                                                                                                                                                                                          | 3       |                                                                  |            |
| <b>5. Metoda programării dinamice: aplicații (1)</b><br>5.1. Problema subsecvenței de sumă maximă. Discutarea diverselor soluții algoritmice și analiza complexității acestora.<br>5.2. Rezolvarea eficientă a problemelor de natură algoritmică folosind structuri de tip Lookup Table (LUT) sau Summed Area Table (SAT).                                                                                                                                                                                                                                                                                                                                                            | 3       |                                                                  |            |
| <b>6. Metoda programării dinamice: aplicații (2)</b><br>6.1. Problema celei mai lungi subsecvențe crescătoare (LIS/LAS) și aplicații privind rezolvarea de probleme asemănătoare întâlnite în practică.<br>6.2. Problema celei mai lungi subsecvențe de elemente comune (LCS) și aplicații privind rezolvarea de probleme asemănătoare întâlnite în practică. Analiza complexității algoritmice a soluțiilor problemelor prezentate.                                                                                                                                                                                                                                                  | 3       | Expunerea, prelegerea, conversația, exemplificarea, demonstrația |            |
| <b>7. Metoda programării dinamice: aplicații (3)</b><br>7.1. Algoritmi eficienți pentru determinarea drumurilor optime în probleme de parcurgere a unui labirint.<br>7.2. Rezolvarea problemelor de drumuri în labirint și prezentarea de aplicații privind probleme asemănătoare întâlnite în practică. Analiza complexității algoritmice a soluțiilor prezentate.                                                                                                                                                                                                                                                                                                                   | 3       |                                                                  |            |
| <b>8. Algoritmi în grafuri: introducere</b><br>8.1. Modalități de reprezentare a grafurilor: matricea de adiacență și liste de adiacență. Exemple.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | 3       |                                                                  |            |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                            |   |  |  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|--|--|
| 8.2. Parcurgerea informației dintr-un graf: algoritmul parcurgerii în lățime (BFS) și algoritmul parcurgerii în adâncime (DFS).                                                                                                                                                                                                                                                                                                                            |   |  |  |
| <b>9. Algoritmi în grafuri: componente conexe</b><br>9.1. Probleme practice rezolvate folosind arborele parcurgerii în adâncime.<br>9.2. Determinarea componentelor conexe ale unui graf neorientat. Analiza complexității algoritmice. Rezolvarea de probleme exemplu.                                                                                                                                                                                    | 2 |  |  |
| <b>10. Algoritmi în grafuri: drumuri</b><br>10.1.Drumuri în graf. Drumuri de cost minim.<br>10.2.Algoritmul Roy-Floyd-Warshall pentru determinarea tuturor drumurilor de cost minim dintre nodurile unui graf.<br>10.3.Algoritmul lui Disjkstra.<br>10.4.Aplicații ale determinării drumurilor într-un graf pentru rezolvarea de probleme specifice.                                                                                                       | 2 |  |  |
| Bibliografie minimală recomandată                                                                                                                                                                                                                                                                                                                                                                                                                          |   |  |  |
| 1. Thomas H. Cormen, Charles E. Leiserson, Ronald R. Rivest, Clifford Stein. 2022. Introduction to Algorithms, 4th Ed. MIT Press (sau traducerea în limba română: Introducere în algoritmi, Ed. Agora)<br>2. Gayle Laakmann McDowell. 2015. Cracking the Coding Interview: 189 Programming Questions and Solutions, 6th Ed., CareerCup<br>3. Radu-Daniel Vatavu. 2013. Ghid practic pentru analiza si proiectarea algoritmilor. Aplicații în C#, MatrixRom |   |  |  |

| Aplicații (Seminar/laborator/lucrări practice/proiect)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | Nr. ore | Metode de predare                                           | Observații                                                                                   |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------|-------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| 1. Analiza complexității algoritmilor (1)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | 2       | Lucrări practice, experimentul, conversația, studiul de caz |                                                                                              |
| 2. Analiza complexității algoritmilor (2)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | 2       |                                                             |                                                                                              |
| 3. Metoda Divide et Impera: Introducere                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 2       |                                                             |                                                                                              |
| 4. Metoda Divide et Impera: Algoritmul sortării prin interclasare. Algoritmul sortării rapide                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 2       |                                                             |                                                                                              |
| 5. Test practic și evaluare                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 2       | Probă practică                                              | Rezolvarea unei probleme algoritmice prin implementarea corectă într-un limbaj de programare |
| 6. Metoda programării dinamice: Introducere. Subsecvența crescătoare de lungime maximă                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | 2       | Lucrări practice, experimentul, conversația, studiul de caz |                                                                                              |
| 7. Metoda programării dinamice: Subsecvența comună de lungime maximă                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 2       |                                                             |                                                                                              |
| 8. Metoda programării dinamice: Parcurgerea labirintului                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | 2       |                                                             |                                                                                              |
| 9. Test practic și evaluare                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | 2       | Probă practică                                              | Rezolvarea unei probleme algoritmice prin implementarea corectă într-un limbaj de programare |
| 10. Grafuri: Introducere. Probleme de reprezentare                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 2       | Lucrări practice, experimentul, conversația, studiul de caz |                                                                                              |
| 11. Grafuri: Parcurgere și componente conexe                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 2       |                                                             |                                                                                              |
| 12. Grafuri: Drumuri de cost minim                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | 2       |                                                             |                                                                                              |
| 13. Test practic și evaluare                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 2       | Probă practică                                              | Rezolvarea unei probleme algoritmice prin implementarea corectă într-un limbaj de programare |
| 14. Rezolvarea de probleme algoritmice specifice                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | 2       | Lucrări practice, test practic, experimentul                |                                                                                              |
| Prezența la activitățile obligatorii este reglementată de “Regulamentul cadru privind evaluarea studenților”. Recuperarea activităților aplicative obligatorii programate pe parcursul semestrului se face în conformitate cu precizările acestui regulament. De asemenea, în conformitate cu prevederile articolului 5.2.3 ale regulamentului sus-menționat, activitățile aplicative programate pe parcursul semestrului pot fi echivalate cu realizarea și susținerea de proiecte de complexitate ridicată din tematica disciplinei, cu acordul cadrului didactic titular. |         |                                                             |                                                                                              |
| Bibliografie minimală recomandată                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |         |                                                             |                                                                                              |
| 1. Thomas H. Cormen, Charles E. Leiserson, Ronald R. Rivest, Clifford Stein. 2022. Introduction to Algorithms, 4th Ed. MIT Press (sau traducerea în limba română: Introducere în algoritmi, Ed. Agora)<br>2. Gayle Laakmann McDowell. 2015. Cracking the Coding Interview: 189 Programming Questions and Solutions, 6th Ed., CareerCup<br>3. Radu-Daniel Vatavu. 2013. Ghid practic pentru analiza si proiectarea algoritmilor. Aplicații în C#, MatrixRom                                                                                                                   |         |                                                             |                                                                                              |

## 8. Evaluare

| Tip activitate             | Criterii de evaluare                                                                                                                                                                                                                                                                                                                                                                                                                      | Metode de evaluare                                                                                                                                                                                               | Pondere din nota finală |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|
| Curs                       | Expunerea clară a conceptelor teoretice și corectitudinea operării cu noțiuni specifice proiectării algoritmilor ca fundamente științifice, ingineresti și ale informaticii. Rezolvarea corectă a problemelor de natură algoritmică pentru prelucrarea datelor în diverse reprezentări și pentru diverse aplicații. Analiza corectă a complexității teoretice a algoritmilor în vederea îmbunătățirii performanțelor sistemelor software. | Probă mixtă, alcătuită din:<br>- lucrare scrisă și<br>- test practic pe calculator (rezolvarea unei probleme de natură algoritmică) cu verificarea orală a abordării folosite pentru rezolvarea testului practic | 50%                     |
| Seminar                    | -                                                                                                                                                                                                                                                                                                                                                                                                                                         | -                                                                                                                                                                                                                | -                       |
| Laborator/lucrări practice | Corectitudinea rezolvării de probleme de natură algoritmică și a implementării acestora folosind un limbaj de programare și o platformă de dezvoltare a aplicațiilor software. Gradul de îndeplinire a activităților din lucrările practice de laborator și realizarea de teme suplimentare acestora. Participarea la diverse activități practice în legătură cu obiectivele specifice ale disciplinei.                                   | Probe practice pe calculator, observația sistematică, verificarea temelor.                                                                                                                                       | 50%                     |
| Proiect                    | -                                                                                                                                                                                                                                                                                                                                                                                                                                         | -                                                                                                                                                                                                                | -                       |

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

| Data completării | Grad didactic, nume, prenume, semnătura titularului de curs | Grad didactic, nume, prenume, semnătura titularului de aplicație |
|------------------|-------------------------------------------------------------|------------------------------------------------------------------|
| 24.09.2025       | Prof. dr. ing. VATAVU Radu-Daniel                           | ș.l. dr. ing. ȘIEAN Alexandru-Ionuț                              |

| Data avizării | Grad didactic, nume, prenume, semnătura responsabilului de program |
|---------------|--------------------------------------------------------------------|
| 25.09.2025    | ș.l. dr. inf. BĂRÎLĂ Adina-Luminița                                |

| Data avizării în departament | Grad didactic, nume, prenume, semnătura directorului de departament |
|------------------------------|---------------------------------------------------------------------|
| 25.09.2025                   | Prof. dr. ing. Ovidiu-Andrei SCHIPOR                                |

| Data aprobării în consiliul facultății | Grad didactic, nume, prenume, semnătura decanului |
|----------------------------------------|---------------------------------------------------|
| 26.09.2025                             | Prof. dr. ing. Laurentiu- Dan MILICI              |