**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**

**Facultatea de Electronică, Telecomunicaţii şi**

**Tehnologia Informaţiei**

# FIŞA DISCIPLINEI

## 1. Date despre program

| | |
|---|---|
| 1.1 Instituţia de învăţământ superior | **Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti** |
| 1.2 Facultatea | **Electronică, Telecomunicaţii şi Tehnologia Informaţiei** |
| 1.3 Departamentul | **Telecomunicaţii** |
| 1.4 Domeniul de studii | Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale |
| 1.5 Ciclul de studii | Licenţă |
| 1.6 Specializarea | Tehnologii şi Sisteme de Telecomunicaţii |

## 2. Date despre disciplină

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2.1 Denumirea disciplinei (ro)<br>(en) | | | | Structuri de date şi algoritmi<br>Algorithms and Data Structures | | | |
| 2.2 Titularul activităţilor de curs | | | | Conf. Dr. Iulian Nastac | | | |
| 2.3 Titularul activităţilor de seminar / laborator | | | | Conf. Dr. Iulian Nastac | | | |
| 2.4 Anul de studiu | 2 | 2.5 Semestrul | 1 | 2.6. Tipul de evaluare | E | 2.7 Regimul disciplinei | Ob |
| 2.8 Tipul disciplinei | | D | 2.9 Codul disciplinei | 04.D.03.O.005 | | 2.10 Tipul de notare | Nota |

## 3. Timpul total estimat (ore pe semestru al activităţilor didactice)

| | | | | | |
|---|---|---|---|---|---|
| 3.1 Număr de ore pe săptămână | 3.5 | Din care: 3.2 curs | 2 | 3.3 seminar/laborator | 1.5 |
| 3.4 Total ore din planul de învăţământ | 49 | Din care: 3.5 curs | 28 | 3.6 seminar/laborator | 21 |
| Distribuţia fondului de timp: | | | | | ore |
| Studiul după manual, suport de curs, bibliografie şi notiţe<br>Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate<br>Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri | | | | | 45 |
| Tutorat | | | | | 0 |
| Examinări | | | | | 6 |
| Alte activităţi (dacă există): | | | | | 0 |

| | |
|---|---|
| 3.7 Total ore studiu individual | 51.00 |
| 3.8 Total ore pe semestru | 100 |
| 3.9 Numărul de credite | 4 |

## 4. Precondiţii (acolo unde este cazul)

| | |
|---|---|
| 4.1 de curriculum | Programarea calculatoarelor şi limbaje de programare |
| 4.2 de rezultate ale învăţării | Capacitatea de a scrie un program de bază în limbajul C/C++. |

## 5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

| | |
|---|---|
| 5.1 Curs | Cursul se va desfăşura într-o sală dotată cu videoproiector şi computer. |

| | |
|---|---|
| 5.2 Seminar/ Laborator/Proiect | Laborator echipat cu calculatoare si software adecvat. Prezența obligatorie la lucrările de laborator (conform regulamentului studiilor universitare de licență în UPB). Accesul individual la sisteme PC dotate cu editoare şi compilatoare C/C++ cât şi cu conexiune la Internet (acces la platforma de e-learning Moodle). |

**6. Obiectiv general** *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învăţământ al specializării studiate etc.)*

Curs: însuşirea mecanismelor de stocare, structurare şi prelucrare a datelor cu alcătuire complexă. Studiul principiilor de bază utilizate în alcătuirea algoritmilor ca etapă esenţială în dezvoltarea eficientă a aplicaţiilor software. Criterii de proiectare eficientă a programelor. Studii de caz şi metode de evaluare a performanţelor algoritmilor. Laborator: însuşirea practică, prin implementare de programe software, a noţiunilor predate la curs. Rezolvarea unor probleme practice concrete care includ elemente de structuri de date şi algoritmi.

**7. Competenţe** *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

| | |
|---|---|
| **Specifice** | Aplicarea cunoştinţelor, conceptelor şi metodelor fundamentale privitoare la arhitectura sistemelor de calcul, microcontrolere, limbaje şi tehnici de programare. |
| **Transversale (generale)** | Capacitatea de a se informa şi documenta permanent pentru dezvoltarea personală şi profesională prin citirea literaturii de specialitate. |

**8. Rezultatele învăţării** *(Sunt enunţuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învăţării reflectă realizările studentului şi mai puţin intenţiile profesorului. Rezultatele învăţării informează studenţii despre ceea ce se aşteaptă de la ei din punct de vedere al performanţei, pentru a obţine notele şi creditele dorite. Sunt definite în termeni concreţi, folosind verbe similare exemplelor de mai jos şi indică ceea ce se va urmări prin evaluare. Rezultatele învăţării vor fi astfel redactate încât să fie evidenţiată clar relaţia faţă de competenţele definite la punctul 7.)*

| | |
|---|---|
| **Cunoştinţe** | *Rezultatul asimilării de informaţii prin învăţare. Cunoştinţele reprezintă ansamblul de fapte, principii, teorii şi practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice şi/sau faptice.*<br>-însuşirea conceptelor avansate de programarea calculatoarelor,<br>-însuşirea programării avansate în limbajul de programare C/C++,<br>-însuşirea cunoştinţelor de a rezolva o problemă avansată de programare,<br>-însuşirea cunoştinţelor de a lucra cu structuri de date avansate,<br>-însuşirea cunoştinţelor de a dezvolta algoritmi avansaţi de programare,<br>-însuşirea cunoştinţelor de a traduce o problemă formulată într-un limbaj natural într-un algoritm şi implementarea programului aferent. |

| | |
|---|---|
| **Aptitudini** | *Capacitatea de a aplica cunoștințe și de a utiliza know-how pentru a duce la îndeplinire sarcini și a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive și creative) sau practice (implicând dexteritate manuală și utilizarea de metode, materiale, unelte și instrumente).*<br>-abilitatea de a întelege şi explica un program scris în cod C/C++,<br>-abilitatea de a de concepe algoritmi avansaţi,<br>-abilitatea de a valida rezultatele unui program C/C++,<br>-abilitatea de a identifica soluţii avansate de programare,<br>-abilitatea de a comunica şi argumenta soluţii. |
| **Responsabilitate și autonomie** | *Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale.*<br>-capacitatea de a selecta şi parcurge surse bibliografice,<br>-capacitatea de a promova şi contribui cu soluţii noi,<br>-capacitatea de a învăţa concepte noi,<br>-capacitatea de comunicare a informaţiilor cu alţi colegi,<br>-dezvoltarea autonomiei în procesul de învățare. |

**9. Metode de predare** *(Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenților la stabilirea propriului parcurs de învățare, cum se identifică eventualele rămâneri în urmă și ce măsuri remediale se adoptă în astfel de cazuri.)*

Curs: Predarea se realizează interactiv folosind atât sistem de videoproiecţie şi prezentări powerpoint dar şi tabla clasică. Sunt prezentate noţiunile de bază şi apoi sunt discutate probleme exemplu. Acestea sunt rezolvate interactiv, parcurgând etapele de înţelegere a cerinţelor, formalizarea algoritmului, elaborarea codului, corectarea erorilor şi validarea rezultatelor. Toate materialele de curs sunt disponibile pe platforma Moodle, în format electronic.
Laborator: Laboratorul se bazează pe programarea individuală în limbajul C/C++ pe platforma Moodle. Fiecare student dispune de un calculator individual. Studenţii au la dispoziţie atât probleme rezolvate, cât şi propuse spre rezolvare. Laboratorul este precedat de scurte prezentări pentru familiarizarea cu conceptele teoretice. Toate materialele de laborator sunt disponibile pe platforma Moodle, în format electronic.

**10. Conţinuturi**

| **CURS** | | |
|---|---|---|
| **Capitolul** | **Conţinutul** | **Nr. ore** |
| 1 | Introducere. Noţiuni generale privind algoritmii. | 1 |
| 2 | Pointeri şi structuri de date: lucrul cu variabile de tip pointer; alocarea dinamică a memoriei (utilizarea reprezentărilor dinamice vs. statice); lucrul cu structuri şi uniuni; exemple, aplicaţii practice. | 3 |
| 3 | Liste de date: definirea şi lucrul cu liste de date; exemple, aplicaţii practice. Cozi de date: definirea şi lucrul cu cozi; exemple, aplicaţii practice. Stive de date: definirea şi lucrul cu stive; exemple, aplicaţii practice. Liste circulare, aplicaţii practice. | 6 |
| 4 | Arbori de date: generalităţi - abordarea unor tipuri de aplicaţii prin intermediul arborilor; definirea şi lucrul cu arbori binari; exemple, aplicaţii practice. | 6 |
| 5 | Arbori speciali, arbori heap. Transformări arbori – vectori. | 4 |
| 6 | lgoritmi de sortare. Aplicaţii practice. Rezolvarea interactivă de probleme diverse ce presupun algoritmizare: pornind de la un enunţ, realizat într-un limbaj formal, se face transformarea acestuia în algoritm şi apoi în program funcţional. | 8 |

| | | |
|---|---|---|
| | **Total:** | 28 |

**Bibliografie:**

D.I. Năstac, Note de curs Programarea Calculatoarelor, UPB, online, platforma Moodle ETTI
C++, http://www.cplusplus.com
D.I. Năstac, Structuri de date și algoritmi – Aplicații, Editura Printech, Bucureşti.

**LABORATOR**

| Nr. crt. | Conţinutul | Nr. ore |
|---|---|---|
| 1 | Pointeri. Structuri de date. | 2 |
| 2 | Liste înlănţuite. | 2 |
| 3 | Stiva. | 2 |
| 4 | Coada. | 2 |
| 5 | Lista circulară | 2 |
| 6 | Arbori binari. | 2 |
| 7 | Aplicaţii algoritmi de sortare/ căutare | 2 |
| 8 | Probleme recapitulative | 3 |
| 9 | Evaluarea unui proiect/temă de casă. | 2 |
| 10 | Colocviu final. | 2 |
| | **Total:** | 21 |

**Bibliografie:**

D.I. Năstac, Note de curs Programarea Calculatoarelor, UPB, online, platforma Moodle ETTI
C++, http://www.cplusplus.com
D.I. Năstac, Structuri de date și algoritmi – Aplicații, Editura Printech, Bucureşti.

**11. Evaluare**

| Tip activitate | 11.1 Criterii de evaluare | 11.2 Metode de evaluare | 11.3 Pondere din nota finală |
|---|---|---|---|
| 11.4 Curs | Cunoaşterea noţiunilor teoretice fundamentale | Examinare în săptămâna 14. Subiectele acoperă întreaga materie. | 40% |
| 11.5 Seminar/laborator/proiect | Verificarea însuşirii cunoștințelor de la laborator | Colocviu final de laborator (oral și la calculator), axat în special pe componenta practică. În plus se vor da teste de laborator, urmărindu-se componenta aplicativă, de deprinderi practice. | 30% |
| | Evaluarea un proiect (temă de casă) | Se urmărește componenta aplicativă, de deprinderi practice. | 30% |
| 11.6 Condiții de promovare | | | |
| - participarea la lucrările de laborator;<br>- cumularea a minim 50% din punctajul aferent disciplinei (laborator și examen). | | | |

**12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)**

Programa de curs oferă absolvenţilor capacitatea de a algoritmiza o problemă de calcul de complexitate medie, cunoaşterea unor algoritmi clasici de soluţionare a problemelor de calcul, capacitatea de a transforma o problemă de calcul exprimată în limbaj natural într-un limbaj de programare şi soluţionarea acesteia. Progresul tehnologic actual al dispozitivelor electronice și de telecomunicații este condiționat de abilitatea de dezvoltare și experimentare folosind limbaje de programare. Astfel că disciplina de programarea calculatoarelor este fundamentală în formarea viitoarelor generații de ingineri și cercetători din domeniu.
Programa asigură astfel absolvenţilor competenţe adecvate cu necesităţile calificărilor actuale și o pregătire ştiinţifică şi tehnică modernă, de calitate şi competitivă, care să le permită angajarea rapidă după absolvire. Aceasta este perfect încadrată în politica Universităţii Politehnica din Bucureşti, atât din punctul de vedere al conţinutului şi structurii, cât şi din punctul de vedere al aptitudinilor şi deschiderii internaţionale oferite studenţilor. Posibilii angajatori vizează atât mediul academic (profil didactic şi de cercetare) cât şi mediul industrial de cercetare-dezvoltare precum organizaţii/firme de orice dimensiune, de la cele mici create de studenţi/masteranzi (de exemplu start-up şi spin-off), până la cele multinaţionale.

| Data completării | Titular de curs | Titular(i) de aplicații |
|---|---|---|
| 21.09.2026 | Conf. Dr. Iulian Nastac | Conf. Dr. Iulian Nastac |

| Data avizării în departament | Director de departament |
|---|---|
| 22.09.2026 | Conf. Dr. Serban Georgica Obreja |

| Data aprobării în Consiliul Facultății | Decan |
|---|---|
| 22.09.2026 | Prof. Dr. Mihnea Udrea |