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	Rețele și Software de Telecomunicaţii

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Structuri de date şi algoritmi			
2.2 Titularul activităţilor de curs				S.l./Lect. Dr. Radu Lupu			
2.3 Titularul activităţilor de seminar / laborator				S.l./Lect. Dr. Radu Lupu			
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 Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					47
Tutorat					0
Examinări					4
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	Parcurgerea următoarelor discipline: • Programarea calculatoarelor si limbaje de programare 1, 2
4.2 de rezultate ale învăţării	Acumularea următoarelor cunoştinţe: • algoritmizarea problemelor • structura si realizarea unui program in 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 calculator.
5.2 Seminar/ Laborator/Proiect	Laboratorul se va desfăşura într-o sală cu dotare specifică, care trebuie să includă: calculatoare desktop/laptop si software instalat: WSL2, Debian/Ubuntu, Chrome/Mozzila, CLion (optional), Compilator C/C++ (ex. gcc), Moodle VPL, access internet (ex. la serverul Gist si 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.)*

Această disciplină se studiază în cadrul domeniului Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale /specializării Reţele şi software de telecomunicaţii şi îşi propune să familiarizeze studenţii cu principalele abordări, modele şi teorii explicative ale domeniului, utilizate în rezolvarea de aplicaţii practice şi probleme, cu relevanţă pentru stimularea procesului de învăţare la studenţi.
Disciplina abordează ca tematică specifică noţiuni de bază/avansate, concepte şi principii specifice, in domeniul calculatoarelor (componentele calculatorului si interactiunea dintre acestea), limbajelor de programare (algoritmizare, descrierea formala a programelor) si, in particular, limbajului C (modelul unui program C, sintaxa limbajului C, functii de biblioteca C) toate acestea contribuind la transmiterea/formarea către/la studenţi a unei viziuni de ansamblu asupra reperelor metodologice şi procedurale aferente domeniului

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	Demonstrează că deţine cunoştinţe de bază/avansate în domeniul Inginerie Electronică, Telecomunicaţii şi Tehnologii Informaţionale Corelează cunoştinţele de programare pentru rezolvarea unor probleme cu ajutorul calculatorului in mod eficient ca timp de executie si resurse necesare. Aplică în practică cunoştinţele de algoritmizare si limbaje de programare prin crearea unor programe in limbaj C cu toate etapele necesare: descriere formala, scriere cod, testare si depanare. Aplică metode şi instrumente standardizate, specifice domeniului, pentru realizarea procesului de evaluare şi diagnoză a unei situaţii, în funcţie de problemele identificate/raportate, şi identifică soluţii. Argumentează si analizează coerent şi corect contextul de aplicare a cunostinţelor de bază ale domeniului, utilizând concepte cheie ale disciplinei şi metodologia specifică. Comunicare orală şi în scris în limba română: utilizează vocabularul ştiinţific specific domeniului, în vederea comunicării eficace, în scris şi oral. Comunicare orală şi în scris într-o limba străină (engleză): demonstrează înţelegerea vocabularului aferent domeniului, într-o limbă străină.

Transversale (generale)	Lucrează în echipă şi comunică eficient, coordonându-şi eforturile cu ceilalţi pentru rezolvarea de situaţii problemă de complexitate medie. Autonomie şi gândire critică: abilitatea de a gândi în termeni ştiinţifici, de a căuta şi analiza date în mod independent, precum şi de a desprinde şi prezenta concluzii / identifica soluţii. Capacitate de analiză şi sinteză: prezintă în mod sintetic cunoştinţele dobândite, ca urmare a unui proces de analiză sistematică. Respectă principiile de etică academică: în activitatea de documentare citează corect sursele bibliografice utilizate. Pune în practică elemente de inteligenţă emoţională în gestionarea socio-emoţională adecvată a unor situaţii din viaţa reală/academică/profesională, demonstrând stăpânire de sine şi obiectivitate în luarea deciziilor sau în situaţii de stres.

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.* Crearea abilităţilor de a aplica cunoştinţele generale privind stucturile de date si structurarea algoritmica a programelor. Posibilitatea de a evalua eficienta implementarii unei aplicatii software folosind criteriile de performanţă analizate la curs. Dezvoltarea calitatilor necesare unui programator: abstractizare, analiză, si sinteză: combinare (abstracţie algebrică), echivalenţă (abstracţie topologică/ sinteză), izolare (descompunere/ analiză), evidenţiere (extragere din context), evidenţiere a unei stări (abstracţie temporală), idealizare (abstracţie de ordine).
Aptitudini	*Capacitatea de a aplica cunoştinţe şi de a utiliza know-how pentru a duce la îndeplinire sarcini şi a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive şi creative) sau practice (implicând dexteritate manuală şi utilizarea de metode, materiale, unelte şi instrumente).* Selectează şi grupează informaţii relevante într-un context dat. •Utilizează argumentat principii specifice în vederea rezolvarii cu ajutorul unui program a unor probleme diverse. •Lucrează productiv în echipă. •Elaborează un text ştiinţific. •Verifică experimental soluţii identificate. •Rezolvă aplicaţii practice. •Interpretează adecvat relaţii de cauzalitate. •Analizează şi compară diverse moduri de rezolvare a unei probleme •Identifică soluţii şi elaborează planuri de rezolvare. •Formulează concluzii la problemele realizate. •Argumentează soluţiile identificate si modurile de rezolvare.

Responsabilitate și autonomie	*Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale.* Selectează surse bibliografice potrivite şi le analizează. •Respectă principiile de etică academică, citând corect sursele bibliografice utilizate. •Demonstrează receptivitate pentru contexte noi de învățare. •Manifestă colaborare cu ceilalți colegi și cadre didactice în desfășurarea activităților didactice •Demonstrează autonomie în organizarea situației/contextului de învățare sau a situației problemă de rezolvat •Manifestă responsabilitate socială prin implicarea activă în viața socială studențească/implicare în evenimentele din comunitatea academică •Promovează/contribuie prin soluții noi, aferente domeniului de specialitate pentru a îmbunătăți calitatea vieții sociale. •Conștientizează valoarea contribuției sale în domeniul ingineriei la identificarea de soluții viabile/sustenabile care să rezolve probleme din viața socială și economică (responsabilitate socială). •Aplică principii de etică/deontologie profesională în analiza impactului tehnologic al soluțiilor propuse în domeniul de specialitate asupra mediului înconjurător. •Analizează și valorifică oportunități de afaceri/de dezvoltare antreprenorială în domeniul de specialitate. •Demonstrează abilități de management al situațiilor din viața reală (gestionarea timpului colaborare vs. conflict).

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.)*

Pornindu-se de la analiza caracteristicilor de învăţare ale studenţilor şi de la nevoile lor specifice, procesul de predare va explora metode de predare atât expozitive (prelegerea, expunerea), cât şi conversative-interactive, bazate pe modele de învăţare prin descoperire facilitate de explorarea directa şi indirectă a realităţii (experimentul, demonstraţia, modelarea), dar şi pe metode bazate pe acţiune, precum exerciţiul, activităţile practice şi rezolvarea de probleme.
În activitatea de predare vor fi utilizate prelegeri, în baza unor prezentări care vor fi puse la dispoziţia studenţilor. Fiecare curs va debuta cu recapitularea capitolelor deja parcurse, cu accent asupra noţiunilor parcurse la ultimul curs.
Prezentările utilizează imagini şi scheme, astfel încât informaţiile prezentate să fie uşor de înţeles şi asimilat.
Acestă disciplină acoperă informaţii şi activităţi practice menite să-i sprijine pe studenţi în eforturile de învăţare şi de dezvoltare a unor relaţii optime de colaborare şi comunicare într-un climat favorabil învăţării prin descoperire.
Se va avea în vedere exersarea abilităţilor de ascultare activă şi de comunicare asertivă, precum şi a mecanismelor de construcţie a feedback-ului, ca modalităţi de reglare comportamentală în situaţii diverse şi de adaptare a demersului pedagogic la nevoile de învăţare ale studenţilor.
Se va exersa abilitatea de lucru în echipă pentru rezolvarea diferitelor sarcini de învăţare

10. Conţinuturi

CURS		
Capitolul	**Conţinutul**	**Nr. ore**

1	1.Introducere 1.1. Tipuri de date derivate: structuri, uniuni, pointeri, tablouri 1.2. Paradigme de programare 1.3. Programare structurata 1.4. Recursivitate	2
2	2. Notiuni generale privind structurile de date 2.1. Reprezentari statice sau dinamice 2.2. Functii specifice	2
3	3. Structuri de date inlantuite (liste) 3.1. Liste simplu inlantuite. Liste dublu inlantuite 3.2. Liste liniare. Liste circulare. 3.3. Cazuri particulare: cozi si stive 3.4. Liste generalizate. Prelucarea recurenta a listelor generalizate	6
4	4. Structuri de date ierarhizate (arbori) 4.1. Definitii, proprietati. 4.2. Implementarea arborilor multipli (n-ari) prin liste generalizate. 4.3. Arbori binari; Parcurgerea arborilor binari. 4.4. Arbori de cautare. 4.5. Arbori de selectie. 4.6. Implementarea arborilor multipli (n-ari) prin arbori binari.	6
5	5. Structuri de date relationale (grafuri) 5.1. Definitii, proprietati. 5.2. Reprezentarea grafurilor prin matrice de adiacenta. 5.3. Reprezentarea grafurilor prin liste de adiacenta. 5.4. Parcurgerea grafurilor. 5.5. Algoritmi de cautare a drumurilor in grafuri	4
6	6. Algoritmi. Principii si tehnici de abordare 6.1. Divide et Impera 6.2. Greedy 6.3. Programare dinamica 6.3. Complexitatea algoritmilor	4
7	7. Algoritmi de sortare 7.1. Metode de sortare prin interschimbare. 7.2. Metode de sortare prin insertie. 7.3. Metode de sortare prin selectie.	4
	Total:	28

Bibliografie:

- *Dan Galaţchi, Sorin Zoican, Roxana Zoican,* Limbajul C. Structuri de date şi algoritmi, Editura POLITEHNICA Press, 2004, 280 pagini, ISBN 973-8449-39-1
- Ovidiu Grigore, C. Bigan, Programarea structurilor de date, Ed. Agerpress Typo, Bucureşti, 2001, ISBN 973-85228-03;
- *Iulian Năstac,* Programarea calculatoarelor in limbajul C – Elemente fundamentale, Editura Printech, Bucureşti, 2006, ISBN 973-718-464-5
-*Tudor Niculiu,* Drumul spre Inteligenţă – de la Informatică şi Intelectică prin Simulare Ierarhică, MATRIX ROM, Bucureşti, 2006, ISBN (13) 978-973-755-078-1
- notite curs – platforma moodle

LABORATOR		
Nr. crt.	**Conţinutul**	**Nr. ore**
1	Recursivitate. Structuri de date derivate	3
2	Liste înlănţuite	3
3	Aplicaţii stive si cozi	3
4	Aplicaţii arbori	3
5	Aplicatii grafuri	3
6	Aplicaţie algoritmi de sortare si cautare	3
7	Colocviu final de laborator	3
	Total:	21

Bibliografie:

- *Dan Galaţchi, Sorin Zoican, Roxana Zoican,* Limbajul C. Structuri de date şi algoritmi, Editura POLITEHNICA Press, 2004, 280 pagini, ISBN 973-8449-39-1
- Ovidiu Grigore, C. Bigan, Programarea structurilor de date, Ed. Agerpress Typo, Bucureşti, 2001, ISBN 973-85228-03;
- *Iulian Năstac,* Programarea calculatoarelor in limbajul C – Elemente fundamentale, Editura Printech, Bucureşti, 2006, ISBN 973-718-464-5
-*Tudor Niculiu,* Drumul spre Inteligenţă – de la Informatică şi Intelectică prin Simulare Ierarhică, MATRIX ROM, Bucureşti, 2006, ISBN (13) 978-973-755-078-1
- www.elcom.pub.ro/discipline/sda

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; - cunoaşterea modului de aplicare a teoriei la probleme specifice; - analiza diferenţială a tehnicilor şi metodelor teoretice.	Două teste scrise de verificare, de ponderi egale, în timpul semestrului, susţinute la date fixate la începutul cursului; subiectele acoperă întreaga materie, realizând o sinteză între parcurgerea teoretică comparativă a materiei şi explicitarea prin exerciţii şi probleme.	50%
	- teme de casa prezentate la curs	Evaluare orala pe calculator	10%

11.5 Seminar/laborator/proiect	-cunoaşterea structurilor de date de baza si a tipurilor de algoritmi pentru utilizarea acestora in rezolvarea unei probleme date; -cunoaşterea modului de utilizare a unor unelte pentru dezvoltarea, simularea si depanarea programelor -demonstrarea funcţionării programelor realizate	-Colocviu final de laborator, cuprinzând o componentă teoretică şi o componentă practică. Componenta teoretică este verificată prin test grilă; componenta practică este evaluată prin verificarea modului de rezolvare (implementare, testare, funcţionare) de către student a unei probleme practice; - Test de realizare a unui program in mediul VPL	40%
11.6 Condiții de promovare			
Obținerea a 50% din punctajul alocat examenului final. Obținerea a 50% din punctajul aferent activității pe parcursul semestrului.			

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)

Prin activitățile desfășurate, studenții dezvoltă abilități de a oferi soluții unor probleme și de a propune idei de îmbunătățire a situației existenței în domeniul Inginerie Electronică, Telecomunicații și Tehnologii Informaționale, ramura industrială Retele si software de telecomunicatii
În dezvoltarea conținutului disciplinei s-au avut în vedere cunoștințe descrise de literatura de specialitate si cercetările proprii publicate si prezentate.
Cursul are un conținut similar cursurilor desfășurate de universitatea POLITEHNICA din Bucuresti.
Se are în vedere dezvoltarea abilităților absolventului de a gestiona situaţii practice cu care se poate confrunta în viața reală în scopul creşterii contribuţiei acestuia la îmbunătățirea mediului socio-economic.

Data completării	Titular de curs	Titular(i) de aplicaţii
21.09.2026	S.l./Lect. Dr. Radu Lupu	S.l./Lect. Dr. Radu Lupu

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

Universitatea Națională de Știință și Tehnologie Politehnica București

Facultatea de Electronică, Telecomunicații și

Tehnologia Informației

Data aprobării în Consiliul Facultății

22.09.2026

Decan

Prof. Dr. Mihnea Udrea