

FIȘA DISCIPLINEI¹

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea ² / Departamentul ³	Facultatea de Inginerie Hunedoara
1.3 Catedra	—
1.4 Domeniul de studii (denumire/cod ⁴)	Științe inginerești aplicate / 270
1.5 Ciclul de studii	licență
1.6 Programul de studii (denumire/cod/calificarea)	Informatică industrială / 50 / Inginer

2. Date despre disciplină

2.1 Denumirea disciplinei/Categoria formativă ⁵	Programarea calculatoarelor și limbaje de programare 2 / DF						
2.2 Titularul activităților de curs	Prof. dr. ing. Pănoiu Manuela						
2.3 Titularul activităților aplicative ⁶	Șef lucr. dr. ing. Ghiormez Loredana						
2.4 Anul de studii ⁷	I	2.5 Semestrul	II	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei ⁸	DI

3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate)⁹

3.1 Număr de ore asistate integral/săptămână	4 , format din:	3.2 ore curs	2	3.3 ore seminar/laborator/proiect	2
3.1* Număr total de ore asistate integral/sem.	56 , format din:	3.2* ore curs	28	3.3* ore seminar/laborator/proiect	28
3.4 Număr de ore asistate parțial/săptămână	, format din:	3.5 ore practică		3.6 ore elaborare proiect de diplomă	
3.4* Număr total de ore asistate parțial/semestru	, format din:	3.5* ore practică		3.6* ore elaborare proiect de diplomă	
3.7 Număr de ore activități neasistate/săptămână	4.93 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			1
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			2.93
3.7* Număr total de ore activități neasistate/semestru	69 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			14
		ore studiu individual după manual, suport de curs, bibliografie și notițe			14
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			41
3.8 Total ore/săptămână ¹⁰	8.93				
3.8* Total ore/semestru	125				
3.9 Număr de credite	5				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Nu este cazul
4.2 de competențe	• Operare pe calculator

5. Condiții (acolo unde este cazul)

¹ Formularul corespunde Fișei Disciplinei promovată prin OMECTS 5703/18.12.2011 și cerințelor Standardelor specifice ARACIS valabile începând cu 01.10.2017.

² Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina.

³ Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului.

⁴ Se înscrie codul prevăzut în HG nr.140/16.03.2017 sau în HG similare actualizate anual.

⁵ Disciplina se încadrează potrivit planului de învățământ în una dintre următoarele categorii formative: disciplină fundamentală (DF), disciplină de domeniu (DD), disciplină de specialitate (DS) sau disciplina complementară (DC).

⁶ Prin activități aplicative se înțeleg activitățile de: seminar (S) / laborator (L) / proiect (P) / practică (Pr).

⁷ Anul de studii în care este prevăzută disciplina în planul de învățământ.

⁸ Disciplina poate avea unul din următoarele regimuri: disciplină impusă (DI), disciplină opțională (DO) sau disciplină facultativă (Df).

⁹ Numărul de ore de la rubricile 3.1*, 3.2*,...,3.8* se obțin prin înmulțirea cu 14 (săptămâni) a numărului de ore din rubricile 3.1, 3.2,...., 3.8. Informațiile din rubricile 3.1, 3.4 și 3.7 sunt chei de verificare folosite de ARACIS sub forma: (3.1)+(3.4) ≥ 28 ore/săpt. și (3.8) ≤ 40 ore/săpt.

¹⁰ Numărul total de ore / săptămână se obține prin însumarea numărului de ore de la punctele 3.1, 3.4 și 3.7.

5.1 de desfășurare a cursului	• Sală de curs, Materiale suport: laptop, proiector, tablă
5.2 de desfășurare a activităților practice	• Laborator cu 12-14 calculatoare, tablă

6. Competențe la formarea cărora contribuie disciplina

Competențe specifice	<p>C1</p> <p>C1.1 Identificarea conceptelor de bază proprii științelor ingineresti aplicate;</p> <p>C1.2 Explicarea structurii și funcționării componentelor diferitelor tipuri de echipamente utilizând teorii și instrumente specifice (scheme, modele matematice, fizice, chimice, biologice etc.);</p> <p>C1.3 Aplicarea tehnicilor de proiectare și a principiilor de construcție a componentelor diferitelor tipuri de echipamente specifice domeniului și specializării;</p> <p>C1.4 Utilizarea metodelor de validare a soluțiilor constructive pentru componentele și structurile proiectate;</p> <p>C1.5 Implementarea de aplicații în practica inginerescă din domeniul specializării, folosind fundamente teoretice ale științelor ingineresti aplicate.</p> <p>C 2.</p> <p>C 2.1. Descrierea structurii și a modului de funcționare a sistemelor informatice în general;</p> <p>C 2.2. Explicarea rolului, funcționalității și utilității sistemelor informatice în general și a sistemelor de prelucrare și gestiune a datelor în domeniul specializării;</p> <p>C 2.3. Utilizarea componentelor software ale sistemelor informatice, folosind algoritmi, protocoale, limbaje, structuri de date;</p> <p>C 2.4. Aprecierea caracteristicilor și calității sistemelor informatice;</p> <p>C 2.5. Prelucrarea și gestionarea datelor utilizând sisteme informatice dedicate.</p> <p>•</p>
Competențele profesionale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • C1 Utilizarea adecvată a fundamentelor teoretice ale științelor ingineresti aplicate • C 2. Utilizarea sistemelor informatice de prelucrare și gestiune a datelor
Competențele transversale în care se înscriu competențele specifice	<ul style="list-style-type: none"> •

7. Obiectivele disciplinei (asociate competențelor de la punctul 6)

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Aprofundarea unui limbaj de programare, limbajul C și deprinderea de a lucra cu structurile de control ale acestui limbaj
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Însușirea modului de lucru cu funcții, înțelegerea structurii unui program, argumentele funcțiilor, tehnicile de transmitere a parametrilor și clasele de memorie.. • Însușirea noțiunilor legate tehnica lucrului cu pointeri; • Însușirea modului de lucru cu fișiere în C. • Deprinderea utilizării funcțiilor de bibliotecă • Însușirea modului de implementare a aplicațiilor complete – tip proiect

8. Conținuturi¹¹

8.1 Curs	Număr de ore	Metode de predare ¹²
----------	--------------	---------------------------------

¹¹ Se detaliază toate activitățile didactice prevăzute prin planul de învățământ (tematicile prelegerilor și ale seminariilor, lista lucrărilor de laborator, conținuturile etapelor de elaborare a proiectelor, tematica fiecărui stagiu de practică). Titlurile lucrărilor de laborator care se efectuează pe standuri vor fi însoțite de notația „(*)”.

1. Instrucțiunile limbajului C 1.1. Instrucțiunea de decizie if; 1.2 Instrucțiunile ciclice while, for, do-while; 1.3. Instrucțiunea switch; 1.4 Instrucțiunile break, continue	4	Expunerea informațiilor esențiale pe videoprojector. Detalierea informațiilor expuse Demonstrarea utilizând un mediu de programare C/C++ Conversația
2. Tipuri structurate 2.1. Tipul tablou 2.2 Tipul structură și union	4	
3. Funcții 3.1. Structura unei funcții. Funcții care returnează o valoare. Funcții care nu returnează o valoare. 3.2. Apelul funcțiilor. Transmiterea parametrilor 3.3. Clase de memorie	4	
4. Pointeri 4.1. Definierea pointerilor și operații cu pointeri 4.2. Prelucrarea tablourilor folosind pointeri	6	
5. Fișiere 5.1. Prelucrarea fișierelor în nivel inferior 5.2. Prelucrarea fișierelor în nivel superior	6	
6. Realizarea unei aplicații tip proiect 6.1.Includeri de fișiere 6.2.Realizarea unui proiect din surse multiple	4	
1. Bibliografie ¹³ Curs format electronic de pe campusul virtual al UPT, https://cv.upt.ro/course/view.php?id=2612 2. Pănoiu M., Muscalagiu I., Pănoiu C. , <i>Utilizarea și programarea calculatoarelor</i> , Editura Mirton, Timișoara, 2002 3. L. Negrescu, <i>Limbajele C și C++ pentru începători</i> , Volumul I – <i>Limbajul C</i> , Editura Alabastră, Cluj-Napoca, 2009 4. V. Iorga, <i>Programare în C</i> , Editura Alabastră, Cluj-Napoca, 2011		
8.2 Activități aplicative¹⁴	Număr de ore	Metode de predare
1. Structura unui program în C/C++. Aplicații cu instrucțiunile alternative ale limbajului C/C++. 2. Aplicații cu instrucțiunile ciclice ale limbajului C.	4	Verificare cunoștințe din tematica laboratorului
3. Declarația de tablou. Aplicații cu tablouri unidimensionale (vectori) 4. Declarația de tablou. Aplicații cu tablouri bidimensionale (matrici)	4	Elaborare aplicații și testare într-un mediu de programare C
5. Funcții. Declarație și apel. Aplicații cu funcții definite de utilizator. 6. Funcții. Declarație și apel. Aplicații cu funcții definite de utilizator. Aplicații cu șiruri de caractere.	4	
7. Pointeri. Aplicații ce utilizează pointeri la prelucrare de tablouri. 8. Pointeri. Aplicații ce utilizează pointeri la prelucrarea șirurilor de caractere.	4	
9. Structuri și tipuri definite de programator. Aplicații cu structuri de date. 10. Pointeri spre structuri. Aplicații	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C
11. Fișiere în limbajul C. Aplicații cu fișiere text. 12. Fișiere în limbajul C. Aplicații cu fișiere binare.	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu

¹² Prezentarea metodelor de predare va include și folosirea noilor tehnologii (e-mail, pagină personalizată de web, resurse în format electronic etc.).

¹³ Cel puțin un un titlu trebuie să aparțină colectivului disciplinei iar cel puțin un titlu trebuie să se refere la o lucrare de referință pentru disciplină, de circulație națională și internațională, existentă în biblioteca UPT.

¹⁴ Tipurile de activități aplicative sunt cele precizate în nota de subsol 5. Dacă disciplina conține mai multe tipuri de activități aplicative atunci ele se trec consecutiv în liniile tabelului de mai jos. Tipul activității se va înscrie într-o linie distinctă sub forma: „Seminar:”, „Laborator:”, „Proiect:” și/sau „Practică:”.

		de programare C
13. Realizarea unui proiect din surse multiple.	4	Verificare cunoștințe Elaborare aplicații și testare într-un mediu de programare C
		Verificare cunoștințe Elaborare aplicații și testare într-un mediu de programare C
Bibliografie ¹⁵ 1. Ghiormez L., Material în format electronic, Campus Virtual UPT, https://cv.upt.ro/course/view.php?id=2612 2. Iordan A., Cristea D., Berdie A., <i>Programarea calculatoarelor</i> , îndrumător de laborator, Editura Mirton, Timișoara, 2004 3. L. Negrescu, <i>Limbajele C și C++ pentru începători</i> , Volumul I – Limbajul C, Editura Alabastră, Cluj-Napoca, 2009		

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

- Disciplina este din categoria disciplinelor fundamentale, și se regăsește în planul de învățământ (eventual sub denumiri apropiate) al tuturor programelor de studii din domeniul fundamental Științe Inginerești din România. Este o disciplină solicitată de toate companiile cu profil IT.

Disciplina se regăsește și în planul de învățământ al programelor de studii din străinătate :

- <https://www.bradley.edu/academic/departments/electrical/programs/ee/>

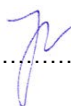
10. Evaluare

Tip activitate	10.1 Criterii de evaluare ¹⁶	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- corectitudinea și completitudinea cunoștințelor asimilate; Studentii trebuie să fie capabili să implementeze programe C	Examen pe campusul virtual cu întrebări cu caracter aplicativ, inclusiv programe C; studenții pot folosi bibliografia, dar fără conexiune la internet	66%
10.5 Activități aplicative	S: L: Prelucrare vectori, matrici, pointeri, fișiere, funcții utilizator P ¹⁷ : Pr:	Verificări periodice pe calculator, lucrare practică la final de semestru	34%
10.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor ¹⁸)			
<ul style="list-style-type: none"> • Să știe să implementeze o aplicație simplă care prelucrează elementele unui vector și să extragă diverse informații din aceste elemente: minim, maxim, suma, contorizări pe diferite condiții, etc 			

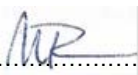
Data completării

05.10.2023

Director de departament
(semnătura)



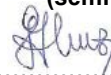
Titular de curs
(semnătura)



Data avizării în Consiliul Facultății¹⁹

16.10.2023

Titular activități aplicative
(semnătura)



Decan
(semnătura)



¹⁵ Cel puțin un titlu trebuie să aparțină colectivului disciplinei.

¹⁶ Fișele disciplinelor trebuie să conțină procedura de evaluare a disciplinei cu precizarea criteriilor, a metodelor și a formelor de evaluare, precum și cu precizarea ponderilor atribuite acestora în nota finală. Criteriile de evaluare se formulează în mod distinct pentru fiecare activitate prevăzută în planul de învățământ (curs, seminar, laborator, proiect). Ele se vor referi și la formele de verificare pe parcurs (teme de casă, referate ș.a.)

¹⁷ În cazul când proiectul nu este o disciplină distinctă, în această rubrică se va preciza și modul în care rezultatul evaluării proiectului condiționează admiterea studentului la evaluarea finală din cadrul disciplinei.

¹⁸ Nu se va explica cum se acorda nota de promovare.

¹⁹ Avizarea este precedată de discutarea punctului de vedere al board-ului de care aparține programul de studii cu privire la fișa disciplinei.