FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea[1] / Departamentul[2]	Inginerie Hunedoara/Inginerie Electrică și Informatică Industrială
1.3 Domeniul de studii (denumire/cod [3])	Inginerie electrică / 90
1.4 Ciclul de studii	Licență
1.5 Programul de studii (denumire/cod/calificarea)	Inginerie electrică și calculatoare / 60 / Inginer

2. Date despre disciplină

2.1a Denumirea disciplinei/Categoria formativă[4]			Programarea calculatoarelor şi limbaje de programare II/DF				
2.1b Denumirea disciplinei în limba engleză			Computer Programming and Programming Languages II				
2.2 Titularul activităţilor de curs			Prof. dr. ing. Pănoiu Manuela				
2.3 Titularul activităţilor aplicative[5]			As. dr. Ing. Rat Cezara				
2.4 Anul de studii[6]	I	**2.5** Semestrul	2	**2.6** Tipul de evaluare	E	**2.7** Regimul disciplinei[7]	DOB

3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate) [8]

3.1 Număr de ore asistate integral/săptămână	5 , format din:	**3.2** ore curs	2	**3.3** ore seminar/laborator/proiect	3
3.1* Număr total de ore asistate integral/sem.	70 , format din:	**3.2*** ore curs	28	**3.3*** ore seminar/laborator/proiect	42
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	3.93 , 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			1.93
3.7* Număr total de ore activități neasistate/ semestru	55 , 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			27
3.8 Total ore/săptămână [9]	8.93				
3.8* Total ore/semestru	125				
3.9 Număr de credite	5				

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	• Nu este cazul
4.2 de rezultatele învățării	•

5. Condiţii (acolo unde este cazul)

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

6. Rezultatele învățării la formarea cărora contribuie disciplina

Cunoștințe	• C1. Studentul/absolventul identifică și descrie concepte, principii și metode de bază din matematică, fizică, economie și informatică. • C8. Studentul/absolventul identifică și descrie principiile utilizate în programare și modelare, simulare, baze de date, inteligență artificială, programare în timp real, învățare automată și rețele neuronale.
Abilități	• A1.Studentul/absolventul ajustează proiectele de produse sau de părți de produse astfel încât acestea să îndeplinească cerințele. Studentul/absolventul creează și/sau execută un plan sau specificație pentru proiectarea unor sisteme industriale, materiale, produse sau un plan de producție, bazate pe concepte de design estetic și/sau funcțional. • A2. Studentul/absolventul specifică proprietăți tehnice ale bunurilor, materialelor, metodelor, proceselor, serviciilor, sistemelor, software-ului și funcționalităților, prin identificarea și răspunsul la nevoile particulare care urmează să fie satisfăcute în funcție de cerințele clienților. • A10. Studentul/absolventul selectează și aplică metodele actuale de modelare, calcul, proiectare și testare pentru specializarea lor.
Responsabilitate și autonomie	• RA1. Studentul/absolventul aplică valorile eticii și deontologiei profesiei de inginer. RA2. Studentul/absolventul practică raționamentul logic, evaluarea și autoevaluare în luarea deciziilor.

7. Obiectivele disciplinei (asociate rezultatelor învățării de la punctul 6)

- Aprofundarea unui limbaj de programare, limbajul C şi deprinderea de a lucra cu structurile de control ale acestui limbaj
  - Însuşirea modului de lucru cu funcţii, înţelegerea structurii unui program, argumentele funcţiilor, tehnicile de transmitere a parametrilor şi clasele de memorie..
  - Însuşirea noţiunilor legate tehnica lucrului cu pointeri;
  - Însuşirea modului de lucru cu fişiere în C.
  - Deprinderea utilizării funcţiilor de bibliotecă
- Însuşirea modului de implementare a aplicaţiilor complete – tip proiect

8. Conţinuturi[10]

8.1 Curs	Număr de ore	Metode de predare[11]
1. Instrucţiunile limbajului C 1.1. Instrucţiunea de decizie if; 1.2 Instrucţiunile ciclice while, for, do-while; 1.3. Instrucţiunea switch; 1.4 Instrucţiunile break, continue	4	Expunerea informațiilor esențiale pe videoproiector. Detalierea informațiilor expuse Demonstrarea utilizând un mediu de programare C/C++ Conversaţia
2. Tipuri structurate 2.1. Tipul tablou 2.2 Tipuri structură și union	4	
3. Funcţii 3.1. Structura unei funcţii. Funcţii care returnează o valoare. Funcţii care nu returnează o valoare. 3.2. Apelul funcţiilor. Transmiterea parametrilor 3.3. Clase de memorie	4	
4. Pointeri 4.1. Definirea pointerilor și operaţii cu pointeri 4.2. Prelucrarea tablourilor folosind pointeri	6	
5. Fişiere 5.1. Prelucarea fişierelor în nivel inferior 5.2. Prelucrarea fişierelor în nivel superior	6	
6. Realizarea unei aplicaţii tip proiect 6.1.Includeri de fişiere	4	

6.2.Realizarea unui proiect din surse multiple		

Bibliografie[12] 1. Panoiu M., Curs în format electronic – Programarea calculatoarelor și limbaje de programare 2,, Campusul virtual al UPTCourse: Programarea calculatoarelor și limbaje de programare 2/Programarea calculatoarelor 2 | CV

2. Pănoiu M., Muscalagiu I., Pănoiu C. , *Utilizarea şi programarea calculatoarelor*, Editura Mirton, Timişoara, 2002
3. L. Negrescu, *Limbajele C şi C++ pentru începători*, Volumul I – *Limbajul C*, Editura Albastră, Cluj-Napoca, 2009
4. V. Iorga, *Programare în C*, Editura Albastră, Cluj-Napoca, 2011

8.2 Activităţi aplicative[13]	Număr de ore	Metode de predare
Aplicaţii cu instrucţiunile ciclice ale limbajului C/C++.	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++
Declaraţia de tablou. Aplicaţii cu tablouri (vectori şi matrici	4	
Aplicaţii cu şiruri de caractere	2	
Funcţii. Declarare şi apel. Aplicaţii cu funcţii definite de utilizator	4	
Pointeri. Aplicaţii ce utilizează pointeri la prelucrare de tablouri	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++
Structuri şi tipuri definite de programator. Aplicaţii cu structuri de date	8	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++
Fişiere în limbajul C. Aplicaţii	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++
Realizare de proiecte în C. Aplicaţii	4	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++
Structuri de date dinamice. Stive. Lista. Coada. Arbori. Aplicații	8	Verificare cunoștințe din tematica laboratorului Elaborare aplicații și testare într-un mediu de programare C/C++

1. Bibliografie[14] 1 Ghiormez L., Pănoiu M., Programarea calculatoarelor în limbajul C/C++, Ed. POLITEHNICA, Timișoara, 326 pg., 2022, ISBN: 978-606-35-0376-4
2. Iordan A., Cristea D., Berdie A., *Programarea calculatoarelor*, îndrumător de laborator, Editura Mirton, Timişoara, 2004
3. L. Negrescu, *Limbajele C şi C++ pentru începători*, Volumul I – Limbajul C, Editura Albastră, Cluj-Napoca, 2009

9. Evaluare

Tip activitate	**9.1** Criterii de evaluare[15]	**9.2** Metode de evaluare	**9.3** Pondere din nota finală
9.4 Curs	- Cunoașterea conceptelor fundamentale ale limbajului C.	Test scris și examen final practic (pe calculator).	66 %

	- Aplicarea corectă a structurilor și funcțiilor în rezolvarea problemelor algoritmice. - Calitatea și funcționalitatea programelor dezvoltate.		
9.5 Activităţi aplicative	**S:**		
	L: - Corectitudinea și funcționalitatea programelor realizate. - Aplicarea corectă a conceptelor limbajului C în rezolvarea problemelor. - Structurarea, claritatea și documentarea codului sursă	Verificări periodice pe calculator, lucrare practică la final de semestru	34%
	P[16]**:**		
	Pr:		
9.6 Standard minim de performanţă (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor[17])			
• Nota de promovare se obține în condițiile obținerii a minim jumătate din punctajul total. • Curs : Realizarea unui program C funcțional care utilizează structuri de control și funcții proprii pentru rezolvarea unei probleme simple • Laborator: Sa promoveze toate testele de laborator			

Data completării 10.09.2026	**Titular de curs** **(semnătura)**	**Titular activităţi aplicative** **(semnătura)**
Director de departament **(semnătura)**	**Data avizării în Consiliul Facultăţii**[18] 23.09.2026	**Decan** **(semnătura)**