

FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea ¹ / Departamentul ²	Electronică, Telecomunicații și Tehnologii Informaționale/Calculatoare și Tehnologia Informației
1.3 Domeniul de studii (denumire/cod ³)	Inginerie electronică, telecomunicații și tehnologii informaționale 20/20/20/200
1.4 Ciclul de studii	Licență
1.5 Programul de studii (denumire/cod/calificarea)	Microelectronică, optoelectronică și nanotehnologii/ 20/20/10/100/40/ Microelectronică, optoelectronică și nanotehnologii

2. Date despre disciplină

2.1 Denumirea disciplinei/Categoria formativă ⁴	Limbaje de Programare 1/DF						
2.2 Titularul activităților de curs	Serafin Petru						
2.3 Titularul activităților aplicative ⁵	Serafin Petru						
2.4 Anul de studii ⁶	1	2.5 Semestrul	1	2.6 Tipul de evaluare	D	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	4 , format din:	3.2 ore curs	2	3.3 ore seminar/laborator/proiect	0/2/0
3.1* Număr total de ore asistate integral/sem.	56 , format din:	3.2* ore curs	28	3.3* ore seminar/laborator/proiect	0/28/0
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	3,14 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			0,5
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1,14
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			1,5
3.7* Număr total de ore activități neasistate/semestru	44 , format din:	ore documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren			7
		ore studiu individual după manual, suport de curs, bibliografie și notițe			16
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			21
3.8 Total ore/săptămână ⁹	7,14				
3.8* Total ore/semestru	100				
3.9 Număr de credite	4				

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	• Nu este cazul
-------------------	-----------------

¹ Se înscrie numele facultății care gestionează programul de studiu căruia îi aparține disciplina.

² Se înscrie numele departamentului căruia i-a fost încredințată susținerea disciplinei și de care aparține titularul cursului.

³ Se înscrie codul prevăzut în HG – privind aprobarea Nomenclatorului domeniilor și al specializărilor/programelor de studii, actualizată anual.

⁴ 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).

⁵ Prin activități aplicative se înțeleg activitățile de: seminar (S) / laborator (L) / proiect (P) / practică (Pr).

⁶ Anul de studii în care este prevăzută disciplina în planul de învățământ.

⁷ Disciplina poate avea unul din următoarele regimuri: disciplină impusă (DI) sau disciplină obligatorie (DOb)-pentru alte domenii fundamentale de studii oferite de UPT, disciplină opțională (DO) sau disciplină facultativă (Df).

⁸ 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.

⁹ Numărul total de ore / săptămână se obține prin însumarea numărului de ore de la punctele 3.1, 3.4 și 3.7.

4.2 de competențe	• Cunoștințe de matematica la nivel de Bacalaureat, competențe digitale de baza
-------------------	---

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	• amfiteatru cu proiector și tablă
5.2 de desfășurare a activităților practice	• laborator cu 15-20 calculatoare, mediu de programare C (Visual Studio) și acces la rețeaua internă CTI/la cv.upt.ro

6. Competențe la formarea cărora contribuie disciplina

Competențe specifice	<ul style="list-style-type: none"> • Operarea cu fundamente științifice, ingineresti și ale informaticii • Identificarea și utilizarea adecvată a principalelor concepte și tehnici de programare • Proiectarea componentelor software și de comunicații • Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor • Deprinderea competențelor de programare în limbajul C
Competențele profesionale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • Sintetizează informații • Interpretează datele actuale • Prezintă rezultatele analizelor • Stabilește procese de date
Competențele transversale în care se înscriu competențele specifice	<ul style="list-style-type: none"> • Efectuează calcule • Aplică cunoștințe științifice, tehnologice și ingineresti

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

7.1 Obiectivul general al disciplinei	<ul style="list-style-type: none"> • Însușirea conceptelor și metodelor de bază privitoare la limbaje și tehnici de programare • Dobândirea noțiunilor de bază din programare, cu exemplificare în limbajul C
7.2 Obiectivele specifice	<ul style="list-style-type: none"> • Dobândirea unei imagini de ansamblu asupra domeniului calculatoarelor și al programării • Proiectarea și implementarea unor programe C de complexitate mică și medie • Obținerea unor deprinderi de testare și depanare a programelor • Însușirea unui stil de programare corect

8. Conținuturi¹⁰

8.1 Curs	Număr de ore	Metode de predare ¹¹
1 Introducere în programarea calculatoarelor 1.1 Programarea sistemelor de calcul 1.2 Scheme logice 1.3 Limbaje de programare 1.4 Scurt istoric al limbajului C	3	Prelegere susținută de prezentări PPT, conversații, explicații, exemplificări
2 Tipuri de date predefinite. Constante, variabile și expresii 2.1 Funcții C 2.2 Tipuri de date în C	3	

¹⁰ 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 „(*)”.

¹¹ Prezentarea metodelor de predare va include și folosirea noilor tehnologii (e-mail, pagină personalizată de web, resurse în format electronic etc.).

2.3 Operatori si expresii in C 2.4 Conversii de tipuri 2.5 Constante numerice 2.6 Variabile		
3 Controlul fluxului de executie. Instructiuni 3.1 Structura unui program C 3.2 Secvențierea 3.3 Operații simple de citire/scriere 3.4 Operatori relaționali 3.5 Operatori logici 3.6 Operatorul condițional 3.7 Instrucțiuni condiționale 3.8 Instrucțiuni de ciclare	3	
4 4 Apeluri de functii. Recursivitate 4.1 Apeluri de functii 4.2 Recursivitate 4.3 Recursivitate vs. iteratie	3	
5 Tablouri 5.1 Variabile locale vs. Variabile globale 5.2 Domenii de vizibilitate pentru variabile 5.3 Tablouri unidimensionale 5.4 Tablouri multidimensionale	3	
6 Pointeri. Siruri de caractere 6.1 Tipul pointer 6.2 Tipul char 6.3 Siruri de caractere 6.4 Functii cu siruri de caractere	3	
7 Pointeri (continuare) 7.1 Tablouri si pointeri 7.2 Aritmetica pointerilor 7.3 7.3 Alocarea dinamica a memoriei 7.4 Pointeri la functii	3	
8 Tipuri definite de utilizator 8.1 Tipul enumerare 8.2 Tipul structura 8.3 Tipul uniune 8.4 Functii de intrare/ iesire	3	
9 Fisiere 9.1 Fisiere text 9.2 Fisiere binare 9.3 Parametri pe linie de comanda 9.4 Functii cu numar variabil de parametrii	3	
10 Reprezentare interna. Operatii pe biti 10.1 Reprezentare interna 10.2 Operatori pe biti 10.3 Preprocesorul C	1	
Bibliografie ¹² Kernighan, B. W., & Ritchie, D. "The C programming language - Second edition", 1988 Prentice Hall Software Series 2. Minea, M., Limbaje de programare, materiale de curs http://staff.cs.upt.ro/~marius/curs/lp/index.html 3. Holotescu, C., Limbaje de programare, materiale de curs http://labs.cs.upt.ro/~oose/pmwiki.php/LP/Lectures 4. Iorga, V., Programarea Calculatoarelor, materiale de curs https://andrei.clubcisco.ro/index.php/anul-1/anul1-sem1/pc		
8.2 Activități aplicative¹³	Număr de ore	Metode de predare

¹² 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.

¹³ 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:”.

Familiarizare cu mediul de programare. Un prim program C de tip „Hello world”	2	Expunere temă cu ajutorul unor materiale auxiliare, exemple, discuții, întrebări, rezolvare pe calculator, a 1-2 probleme.
Tipuri fundamentale. Operatori matematici și de atribuire	2	
Functii operatii simple de citire/scriere	2	
Instructiunea de decizie. Functii recursive	2	
Instructiunea de ciclare. Tablouri (vectori, matrice)	4	
Tipul caracter. Siruri de caractere	4	
Pointeri	4	
Tipuri definite de utilizator	4	
Operatii cu fisiere	4	
Bibliografie ¹⁴ 1. Minea, M., Limbaje de programare, materiale de curs http://staff.cs.upt.ro/~marius/curs/lp/index.html 2. Holotescu, C., Limbaje de programare, materiale de curs http://abs.cs.upt.ro/~oose/pmwiki.php/LP/Lectures 3. Materialele de laborator de Programare a calculatoarelor (OpenCourseWare)		

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

<p>Cunoștințele de programare sunt importante pentru toate materiile cu specific software care fac parte din planul de învățământ al specializării: Programarea și utilizarea calculatoarelor; Programare orientată pe obiecte; Limbaje de programare 2, etc.</p> <p>Majoritatea angajatorilor reprezentativi din domeniul aferent programului solicită atât cunoștințe de programare în general cât și cunoașterea limbajului de programare C.</p> <p>Compatibilitate internațională</p> <ul style="list-style-type: none"> Carnegie Mellon University http://www.ece.cmu.edu/courses/items/18213.html University of Michigan https://www.eecs.umich.edu/eecs/academics/special_topics.html
--

10. Evaluare

Tip activitate	10.1 Criterii de evaluare ¹⁵	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	Înțelegerea conceptelor prezentate la curs și rezolvarea unor aplicații propuse la fiecare capitol de curs.	Verificarea cunoștințelor se efectuează prin examen scris distribuit în trei părți cu subiecte la fiecare parte: Se dau atât subiecte de teorie, prin întrebări de tip grilă, cât și probleme de tipul celor propuse la curs și la laborator în mediul de programare Visual Studio, limbajul de programare C. Partea teoretică este evaluată sub forma unui test grilă pe Campus Virtual cu notarea răspunsurilor scrise pe formular, iar partea aplicativă este evaluată prin implementarea programelor C compilate pe calculator	0.5
10.5 Activități aplicative	S:		
	L: Implementarea de programe și teste grila	Studentii sunt evaluați în timpul laboratoarelor prin teste de tip grilă și prin notarea programelor dezvoltate exclusiv în timpul laboratorului. Se pot primi teme de lucru, care vor fi notate cu punctaje suplimentare	0.5
	P¹⁶:		
	Pr:		
10.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor¹⁷)			
<ul style="list-style-type: none"> Scrierea de cod în limbajul C, program compilabil Proiectarea, testarea și executarea unui program de complexitate medie: 3-4 funcții, transmitere de parametri la funcții Stăpânirea lucrului cu structuri de date: tablouri liniare, șiruri de caractere, structuri Operații de bază în C 			

¹⁴ Cel puțin un titlu trebuie să aparțină colectivului disciplinei.

¹⁵ 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.)

¹⁶ 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.

¹⁷ Nu se va explica cum se acorda nota de promovare.

Data completării

4.12.2024

**Titular de curs
(semnătura)**

Sl. dr. ing. Serafin Petru

**Titular activități aplicative
(semnătura)**

Sl. dr. ing. Serafin Petru

**Director de departament
(semnătura)**

Prof.dr.ing. Mihai-Victor MICEA

Data avizării în Consiliul Facultății¹⁸

10.12.2024

**Decan
(semnătura)**

Prof. dr. ing Cătălin CĂLEANU

¹⁸ Avizarea este precedată de discutarea punctului de vedere al board-ului de care aparține programul de studii cu privire la fișa disciplinei.