FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	Universitatea Politehnica Timișoara
1.2 Facultatea[1] / Departamentul[2]	Inginerie din Hunedoara/Inginerie Electrică și Informatică Industrială
1.3 Domeniul de studii (denumire/cod [3])	IINGINERIA AUTOVEHICULELOR/ 160
1.4 Ciclul de studii	Licență
1.5 Programul de studii (denumire/cod/calificarea)	AUTOVEHICULE RUTIERE / 30 / Inginer

2. Date despre disciplină

2.1a Denumirea disciplinei/Categoria formativă[4]			Programarea calculatoarelor și limbaje de programare / DF				
2.1b Denumirea disciplinei în limba engleză			Computer programming and programming languages				
2.2 Titularul activităţilor de curs			Șef lucr. dr. ing. Berdie Adela Diana				
2.3 Titularul activităţilor aplicative[5]			Șef lucr. dr. ing. Berdie Adela Diana				
2.4 Anul de studii[6]	I	**2.5** Semestrul	I	**2.6** Tipul de evaluare	V	**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			0,71
		ore studiu individual după manual, suport de curs, bibliografie și notițe			1,07
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			2,15
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			10
		ore studiu individual după manual, suport de curs, bibliografie și notițe			15
		ore pregătire seminarii/laboratoare, elaborare teme de casă și referate, portofolii și eseuri			30
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	• Cunoștințe de matematică elementară – nivel liceu
4.2 de rezultatele învățării	• Cunoștințe generale de operare pe calculator: sistem de operare, lucrul cu fișiere/ directoare; • Noțiuni elementare de matematică

5. Condiţii (acolo unde este cazul)

5.1 de desfăşurare a cursului	• Sală de curs echipată cu tabla, videoproiector, calculator și conexiune la Internet. • Predarea in format online a cursului se desfășoară folosind uneltele CVUPT prin intermediul Campusului Virtual (CV) și al aplicației Zoom (videoconferințe) • Toate resursele educaționale utilizate pentru desfășurarea cursurilor sunt încărcate pe platforma CV
5.2 de desfăşurare a activităţilor practice	• Sală de laborator echipată cu computere, tabla, videoproiector și aplicații necesare implementarii de programe cu conexiune la Internet

	• Desfășurarea in format online a activităților practice (laborator) se realizează folosind uneltele CVUPT prin intermediul Campusului Virtual (CV) și al aplicației Zoom (videoconferințe) • Toate resursele educaționale utilizate pentru desfășurarea laboratoarelor sunt încărcate pe platforma CV

6. Rezultatele învățării la formarea cărora contribuie disciplina

Cunoștințe	• C1. Studentul/absolventul identifică și explică conceptele, teoriile și metodele de bază ale domeniului ingineriei autovehiculelor și ale specializării. • C8. Studentul/absolventul recunoaște componentele hardware ale unui sistem digital, utilizează motoare de căutare și aplicații de editare, și este conștient de riscurile mediului digital și de metodele de protecție
Abilități	• A2. Studentul/absolventul aplică principii și metode de bază și rezolvă probleme asociate reprezentărilor grafice, bazelor de date, modelării și simulării sistemelor și proceselor specifice autovehiculelor rutiere • A13. Studentul/absolventul utilizează corect echipamentele digitale, selectează informații relevante, redactează și salvează conținut adaptat scopului, respectând totodată regulile de securitate și protecție a datelor personale.
Responsabilitate și autonomie	• RA3. Studentul/absolventul își asumă responsabilitatea pentru dezvoltarea profesională continuă, folosind surse de informare tehnică de specialitate și tehnici moderne de învățare. • RA8. Studentul/absolventul respectă normele de utilizare a echipamentelor digitale, verifică sursele informaționale, realizează autonom materiale digitale cu respectarea eticii și manifestă responsabilitate în protejarea identității și a datelor.

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

- **Obiectiv principal**: dezvoltarea competențelor fundamentale de algoritmică și programare în limbajul C, ca instrument de modelare, și prelucrare a datelor în ingineria autovehiculelor rutiere, în condițiile utilizării responsabile și sigure a mediului digital
- **Obiective asociate**: La finalizarea părții teoretice (cursuri) și aplicative (laboratoare) studentul va fi capabil să:
  - enumere și descrie conceptele fundamentale ale algoritmicii, structurilor de control (alternative și repetitive) și principiile programării structurate.
  - explice elementele de bază ale vocabularului și sintaxei limbajului C (tipuri de date, operatori, expresii, tablouri, funcții și structuri de date) utilizate în prelucrarea datelor tehnice specifice autovehiculelor rutiere.
  - recunoască componentele hardware ale unui sistem de calcul și funcțiile mediilor integrate de dezvoltare în realizarea de programe (compilare, rulare, depanare)
  - transpună și aplice algoritmi fundamentali în programe executabile scrise în limbajul C, utilizând mediul de programare Code Blocks sau alte medii compatibile
  - selecteze și utilizeze corect structurile de date adecvate (vectori, matrice, șiruri de caractere, structuri) pentru modelarea și rezolvarea numerică a unor probleme specifice domeniului autovehiculelor rutiere.
  - anticipeze erorile de compilare sau de execuție, verificând corectitudinea codului sursă și eficiența acestuia prin rularea de cazuri de test în laborator.
  - selecteze și analizeze surse bibliografice de specialitate, îndrumătoare de laborator și documentație tehnică oficială pentru formarea continuă în domeniul programării.
  - demonstreze autonomie în organizarea mediului de lucru digital și în rezolvarea problemelor practice de laborator, respectând normele de etică academică și evitarea plagiatului.
  - manifeste colaborare activă cu colegii și cadrele didactice în cadrul activităților aplicative, asumându-si responsabilitatea pentru protejarea datelor și utilizarea corectă a echipamentelor informatice.
-

8. Conţinuturi[10]

8.1 Curs	Număr de ore	Metode de predare[11]
1. 1. Introducere în algoritmi 1.1. Definiţie. Proprietăţi 1.2. Obiectele cu care lucrează algoritmii 1.2.1. Date și tipuri de date 1.2.2. Operanzi 1.2.3. Operatori și expresii 1.3. Modalități de reprezentare a algoritmilor 1.3.1. Schema logică 1.3.2. Limbajul pseudocod	2	Expunerea informațiilor esențiale pe videoproiector sau prin ZOOM pentru formatul online Detalierea informațiilor expuse Demonstrarea utilizând un mediu de programare adecvat Conversaţia
2. Principiile programării structurate 2.1. Generalități 2.2. Structuri de bază și descrierea lor 2.2.1. Structura secvențială 2.2.2. Structura alternativă – simplă, dublă, multiplă 2.2.3. Structura repetitivă 2.2.3.1. Ciclul cu număr cunoscut de pași 2.2.3.2. Ciclul cu test inițial 2.2.3.3. Ciclul cu test final	4	
3. Analiza algoritmilor 3.1. Verificarea corectitudinii 3.2. Calculul efectuat prin execuția unui algoritm	2	
4. Limbajul de programare C 4.1 Scurtă introducere 4.2 Structura unui program 4.3 Funcții standard de intrare/ieșire 4.4 Compilare și rulare – noțiuni de bază 4.5 Comentarii 4.6 Exemplu de program	2	
5. Vocabularul limbajului C 5.1 Noțiuni de bază 5.2 Nume de variabile 5.3 Tipuri și dimensiuni de date 5.4 Constante, declarații 5.5 Expresii 5.6 Operatori	4	
6. Controlul fluxului de execuție. Instrucțiuni 6.1 Instrucțiunea expresie 6.2 Instrucțiunea IF 6.3 Instrucțiunea compusă 6.4 Instrucțiunea SWITCH 6.5 Instrucțiunile FOR, WHILE și DO WHILE 6.6 Instrucțiuni de salt – BREAK, CONTINUE, GOTO, RETURN	6	
7. Tablouri. 7.1 Tablouri unidimensionale 7.2 Tablouri bidimensionale 7.3 Șiruri de caractere	4	
8. Funcții și structuri 8.1 Declararea și definirea funcțiilor 8.2 Apelarea funcțiilor. Transmiterea parametrilor prin valoare 8.3 Definirea tipului structură 8.4 Declararea variabilelor de tip structură și inițializarea lor 8.5 Accesul la membrii unei structuri	4	

Bibliografie[12] 1. A.D. Berdie – Programarea calculatoarelor și limbaje de programare, Suport curs – Campus Virtual UPT, 2026
2. Ghiormez Loredana, Pănoiu Manuela – Programarea calculatoarelor în limbajul C/C++, Editura Politehnica, Timișoara, 2022, ISBN 978-606-35-0499-0.
3. Morariu-Gligor Radu Mircea, Moholea Iuliana Fabiola, Șerdean Florina Maria – Programare în limbajul C cu aplicații în inginerie (Vol. I), Editura U.T.PRESS, Cluj-Napoca, 2021, ISBN 978-606-737-550-3
4. Popa Horia Emil, Udrescu Mihai – Programarea calculatoarelor și limbaje de programare. Bazele programării în limbajul C, Editura Politehnica, Timișoara, 2021.
5. Iacob Ioana-Mihaela, Munteanu Radu – Algoritmi și programare în limbajul C. Curs și aplicații, Editura U.T.PRESS, Cluj-Napoca, 2023.
6. Dumitrache Ioan, Călin Sergiu, Moisescu Mihnea Alexandru – Programarea calculatoarelor în limbajul C. Fundamente și aplicații, Editura Printech, București, 2021.
7. Vălean Honoriu, Ungureanu Liviu – Programarea calculatoarelor. Algoritmi și structuri de date în C, Editura Risoprint, Cluj-Napoca, 2022.

8.2 Activităţi aplicative[13]	Număr de ore	Metode de predare
1. Dezvoltarea corectă a algoritmilor din specificații. Evaluarea expresiilor	2	Verificare cunoștințe din tematica laboratoarelor Activitățile aplicative se realizează în limbajul C, cu ajutorul utilitarului Code Bloks. Pentru toate laboratoarele discuții, întrebări și îndrumare în rezolvarea la tablă și pe calculator a aplicațiilor. Pentru formatul ONLINE, videoconferinta ZOOM prin CV UPT
2. Reprezentarea algoritmilor pentru structuri secventiale și alternative	4	
3. Reprezentarea algoritmilor pentru structuri repetitive	4	
4. Verificarea corectitudinii algoritmilor din structurile de control	4	
5. Editarea și testarea aplicațiilor în limbaj C cu ajutorul utilitarului Code Blocks. Programe simple cu funcții de intrare/ieșire standard	2	
6. Aplicații utilizând instrucțiunile IF și SWITCH	4	
7. Utilizarea instrucțiunilor FOR, WHILE și DO WHILE și a instrucțiunilor de salt în aplicații	8	
8. Aplicații cu prelucrări asupra tablourilor unidimensionale, bidimensionale și a șirurilor de caractere	8	
9. Aplicații utilizând funcții și structuri pentru memorarea datelor	6	

Bibliografie[14] 1. A.D. Berdie – Programarea calculatoarelor și limbaje de programare, Suport laborator – CV UPT, 2026
2. Iacob Ioana-Mihaela, Munteanu Radu – Algoritmi și programare în limbajul C. Curs și aplicații, Editura U.T.PRESS, Cluj-Napoca, 2023.
3. Munteanu Radu, Iacob Ioana-Mihaela – Programarea calculatoarelor în limbajul C. Probleme rezolvate și propuse, Editura U.T.PRESS, Cluj-Napoca, 2023
4. Morariu-Gligor Radu Mircea, Moholea Iuliana Fabiola, Șerdean Florina Maria – Programare în limbajul C cu aplicații în inginerie (Vol. II), Indrumător de laborator, Editura U.T.PRESS, Cluj-Napoca, 2021.
5. Pănoiu, Manuela; Ghiormez, Loredana – Programarea calculatoarelor în limbajul C/C++. Îndrumător de laborator, Editura Politehnica, Timișoara, 2022
6. Dumitrache Ioan, Călin Sergiu, Moisescu Mihnea Alexandru – Programarea calculatoarelor în limbajul C. Fundamente și aplicații, Editura Printech, București, 2021

9. Evaluare

Tip activitate	**9.1** Criterii de evaluare[15]	**9.2** Metode de evaluare	**9.3** Pondere din nota finală
9.4 Curs	Corectitudinea și completitudinea cunoștințelor despre algoritmi și programare C; capacitatea de a realiza și interpreta corect un algoritm și a-l transpune într-un	Examen final scris - subiecte teoretice și probleme (două părți de evaluare – algoritmi și limbaj C) În caz de scenariu online se utilizează o aplicație ZOOM cu test grilă pentru teorie, iar rezolvările problemelor se încarcă pe campusul virtual UPT	0,66

	program în limbajul C		
9.5 Activităţi aplicative	**S:**		
	L: Corectitudinea, calitatea codului și justificarea interpretărilor; autonomie în rezolvare Teme de casă și de laborator în timpul desfășurării activității practice	Evaluări pe parcursul semestrului: - Scris – aplicații folosind scheme logice/ limbaj pseudocod și limbaj C. - Oral – aplicații utilizând computerul. In caz de scenariu online testele de evaluare se desfășoară prin intermediul CV-UPT	0,34
	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])

Promovarea disciplinei:

- note de minim 5 la activitatea de laborator şi la examenele scrise pentru cele două părți
- nota finală pe disciplină trebuie să fie minim 5.
- prezență și predare a tuturor temelor de la lucrările de laborator

Cunoștințe minimale pentru promovarea disciplinei:

- Laborator: studentul trebuie să rezolve corect aplicațiile rezolvate prin intermediul algoritmilor și a limbajului de programare C sau să obțină minim 50% din punctajul total alocat subiectelor la evaluările de pe parcurs
- Curs: studentul trebuie să rezolve 50% din partea de teorie și 50% din partea aplicativă care constă în rezolvarea de expresii cu operatori, descrierea problemelor cu ajutorul algoritmilor și implementarea acestora în limbaj C.
-

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