

## FIȘA DISCIPLINEI

### 1. Date despre program

|  |  |
|--|--|
| 1.1 Instituția de învățământ superior                    | Universitatea Politehnica Timișoara                                    |
| 1.2 Facultatea <sup>1</sup> / Departamentul <sup>2</sup> | Inginerie din Hunedoara/Inginerie Electrică și Informatică Industrială |
| 1.3 Domeniul de studii (denumire/cod <sup>3</sup> )      | 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 <sup>4</sup> | Programarea calculatoarelor și limbaje de programare / DF |               |   |                       |   |                                      |    |
| 2.1b Denumirea disciplinei în limba engleză                 | Computer programming and programming languages            |               |   |                       |   |                                      |    |
| 2.2 Titularul activităților de curs                         | Șef lucr. dr. ing. Berdie Adela Diana                     |               |   |                       |   |                                      |    |
| 2.3 Titularul activităților aplicative <sup>5</sup>         | Șef lucr. dr. ing. Berdie Adela Diana                     |               |   |                       |   |                                      |    |
| 2.4 Anul de studii <sup>6</sup>                             | I   | 2.5 Semestrul | 1 | 2.6 Tipul de evaluare | V | 2.7 Regimul disciplinei <sup>7</sup> | DI |

### 3. Timp total estimat - ore pe semestru: activități didactice directe (asistate integral sau asistate parțial) și activități de pregătire individuală (neasistate)<sup>8</sup>

|  |                    |  |    |                                       |      |
|--|--------------------|--|----|---------------------------------------|------|
| 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 <sup>9</sup>                   | 8,93               |  |    |                                       |      |
| 3.8* Total ore/semestru                                | 125                |  |    |                                       |      |
| 3.9 Număr de credite                                   | 5                  |  |    |                                       |      |

### 4. Precondiții (acolo unde este cazul)

|                              |   |
|------------------------------|---|
| 4.1 de curriculum            | • Cunoștințe de matematică elementară – nivel liceu |
| 4.2 de rezultatele învățării | •   |

### 5. Condiții (acolo unde este cazul)

|   |  |
|---|--|
| 5.1 de desfășurare a cursului               | <ul style="list-style-type: none"> <li>• Sală de curs echipată cu videoprojector și conexiune la Internet.</li> <li>• Studenții nu se vor prezenta la prelegeri cu telefoanele mobile deschise.</li> <li>• Nu se acceptă părăsirea sălii de curs fără aprobarea cadrului didactic</li> <li>• Predarea în format online a cursului se desfășoară folosind unelele CVUPT prin intermediul Campusului Virtual (CV) și al aplicației Zoom (videoconferințe)</li> <li>• Toate resursele educaționale utilizate pentru desfășurarea cursurilor sunt încărcate pe platforma CV</li> </ul> |
| 5.2 de desfășurare a activităților practice | <ul style="list-style-type: none"> <li>• Sală de laborator echipată cu computere și aplicații necesare implementării de programe</li> </ul>  |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>• Studenții nu se vor prezenta la activitățile practice cu telefoanele mobile deschise.</li> <li>• Nu se acceptă părăsirea sălii de desfășurare a activității practice fără aprobarea cadrului didactic</li> <li>• Desfășurarea în format online a activităților practice (laborator) se realizează folosind uneltele CVUPT prin intermediul Campusului Virtual (CV) și al aplicației Zoom (videoconferințe)</li> <li>• Toate resursele educaționale utilizate pentru desfășurarea laboratoarelor sunt încărcate pe platforma CV</li> </ul> |
|--|--|

## 6. Rezultatele învățării la formarea cărora contribuie disciplina

|                               |   |
|-------------------------------|---|
| Cunoștințe                    | <ul style="list-style-type: none"> <li>• C1. Studentul/absolventul identifică și explică conceptele, teoriile și metodele de bază ale domeniului ingineriei autovehiculelor și ale specializării.</li> <li>• 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</li> </ul>  |
| Abilități                     | <ul style="list-style-type: none"> <li>• 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</li> <li>• 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.</li> </ul> |
| Responsabilitate și autonomie | <ul style="list-style-type: none"> <li>• RA3. Studentul/absolventul își asumă responsabilitatea pentru dezvoltarea profesională continuă, folosind surse de informare tehnică de specialitate și tehnici moderne de învățare.</li> <li>• 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.</li> </ul>                |

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

|  |
|--|
| <ul style="list-style-type: none"> <li>• Dezvoltarea capacităților studenților de a dezvolta o gândire algoritmică în abordarea rezolvării problemelor ingineresti</li> <li>• Familiarizarea cu un limbaj de programare procedural – limbajul C – și cu paradigmele programării procedurale prin realizarea de aplicații specifice</li> <li>• Dobândirea noțiunilor specifice gândirii algoritmice în scopul soluționării problemelor aplicative</li> <li>• Însușirea principiilor ce stau la baza programării</li> <li>• Însușirea unei gândiri algoritmice și dezvoltarea abilităților de proiectare de soluții algoritmice</li> <li>• Inițiere în utilizarea limbajului de programare C prin implementarea algoritmilor dezvoltați</li> </ul> |
|--|

## 8. Conținuturi<sup>10</sup>

| 8.1 Curs   | Număr de ore | Metode de predare <sup>11</sup>   |
|--|--------------|---|
| 1. Algoritmi<br>1.1 Noțiuni generale<br>1.2 Etapele rezolvării unei probleme – enunț, date intrare/ ieșire<br>1.3 Obiectele cu care lucrează algoritmi și operații permise<br>1.4 Operații pe care le efectuează un algoritm | 2            | Prelegere susținută prin prezentări PPT, conversații, explicații, exemplificări |
| 2. Principiile programării structurate<br>2.1 Generalități<br>2.2 Moduri de reprezentare a algoritmilor<br>2.3 Structuri de bază și descrierea lor   | 4            |   |

|  |                     |   |
|--|---------------------|---|
| 3. Elemente de bază ale limbajului C<br>3.1 Noțiuni introductive<br>3.2 Prezentarea mediului de programare Code Blocks. Primul program<br>3.3 Structura programelor – de la cod sursă la program executabil<br>3.4 Funcții standard de intrare/ieșire  | 2                   |   |
| 4. Vocabularul limbajului C<br>4.1 Noțiuni de bază<br>4.2 Nume de variabile<br>4.3 Tipuri și dimensiuni de date<br>4.4 Constante, declarații<br>4.5 Expresii<br>4.6 Operatori  | 2                   |   |
| 5. Controlul fluxului de execuție. Instrucțiuni<br>5.1 Instrucțiunea expresie<br>5.2 Instrucțiunea IF<br>5.3 Instrucțiunea compusă<br>5.4 Instrucțiunea SWITCH<br>5.5 Instrucțiunile FOR, WHILE și DO WHILE<br>5.6 Instrucțiuni de salt – BREAK, CONTINUE, GOTO, RETURN  | 8                   |   |
| 6. Tablouri.<br>6.1 Tablouri unidimensionale<br>6.2 Tablouri bidimensionale<br>6.3 Șiruri de caractere   | 6                   |   |
| 7. Funcții și structuri<br>7.1 Declararea structurilor<br>7.2 Transmiterea tablourilor ca parametri ai funcțiilor<br>7.3 Transmiterea structurilor și șirurilor ca parametri ai funcțiilor   | 4                   |   |
|  |                     |   |
|  |                     |   |
|  |                     |   |
|  |                     |   |
|  |                     |   |
|  |                     |   |
| Bibliografie <sup>12</sup> 1. A.D. Berdie – Material electronic curs – Campus Virtual UPT, 2025<br>2. Anca Iordan, Daniela Cristea, Adela Berdie, Programarea calculatoarelor, Editura Mirton, 2004<br>3. Brian W. Kernighan, Denis Ritchie, Limbajul C, Ed. Teora, 2003<br>4. Dr. Kris Jamsa & Lars Klander, Totul despre C și C++. Manualul fundamental de programare în C și C++, Ed. Teora, 2001<br>5. Liviu Negrescu, Limbajele C și C++ pentru începători, vol. I, Limbajul C, Ed. MicroInformatica, 1994<br>6. V.Iorga, Programare în C, Ed.Albastră, Cluj Napoca, 2011 |                     |   |
| <b>8.2 Activități aplicative<sup>13</sup></b>  | <b>Număr de ore</b> | <b>Metode de predare</b>  |
| 1. Dezvoltarea corectă a algoritmilor din specificații. Evaluarea expresiilor  | 2                   | Expunere temă, discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor |
| 2. Reprezentarea algoritmilor prin limbaj pseudocod  | 6                   |   |
| 3. Reprezentarea algoritmilor prin scheme logice   | 6                   |   |
| 4. Editarea și testarea aplicațiilor în limbaj C cu ajutorul utilitarului Code Blocks. Programe simple cu funcții de intrare/ieșire standard   | 2                   |   |
| 5. Utilizarea instrucțiunilor IF și SWITCH   | 4                   | Expunere temă, discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor |
| 6. Utilizarea instrucțiunii FOR, WHILE și DO WHILE   | 8                   | Expunere temă,  |

|  |   |   |
|--|---|---|
|  |   | discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor                |
| 7. Prelucrări asupra tablourilor unidimensionale, bidimensionale și a șirurilor de caractere   | 8 | Expunere temă, discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor |
| 8. Definirea și utilizarea funcțiilor și structurilor pentru memorarea datelor   | 4 | Expunere temă, discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor |
| 9. Recapitulare și recuperare  | 2 | Expunere teme, discuții, întrebări, îndrumare în rezolvarea pe calculator a problemelor |
| Bibliografie <sup>14</sup> 1. A. Berdie, Material electronic laborator – Campus Virtual UPT, 2025<br>2. A. Iordan, D. Cristea, A. Berdie, Programarea calculatoarelor, Editura Mirton, Timișoara, 2004<br>3. I. Muscalagiu, D. Muscalagiu, M. Pănoiu, Culegere de probleme în limbajul C, Editura Mirton, Timișoara, 2001<br>4. H. Ciocârlie, R. Ciocârlie, Tehnici de programare și structuri de date, Editura Eurostampa, 2012<br>5. V. Iorga, Programare în C, Editura Albastră, 2011 |   |   |

## 9. Evaluare

| Tip activitate   | 9.1 Criterii de evaluare <sup>15</sup>  | 9.2 Metode de evaluare   | 9.3 Pondere din nota finală |
|--|---|--|-----------------------------|
| 9.4 Curs   | Cunoștințe teoretice – Algoritmi și programare limbajul C   | Scris - subiecte teoretice și aplicații (două părți de evaluare – algoritmi și limbaj C / Test online - platforma CV – (două părți de evaluare –teorie și aplicații probleme ) | 0,66                        |
| 9.5 Activități aplicative  | <b>S:</b>   |  |                             |
|  | <b>L:</b> Abilități în rezolvarea aplicațiilor de laborator – realizarea de algoritmi și programe în limbajul C | Oral -verificare la încheierea activității de laborator / Test online - platforma CV – aplicații cu probleme din algoritmi si limbajul C                                       | 0,34                        |
|  | <b>P<sup>16</sup>:</b>  |  |                             |
|  | <b>Pr:</b>  |  |                             |
| <b>9.6 Standard minim de performanță (se prezintă cunoștințele minim necesare pentru promovarea disciplinei și modul în care se verifică stăpânirea lor<sup>17</sup>)</b>  |   |  |                             |
| <ul style="list-style-type: none"> <li>• Este necesar ca studentul să fie capabil a</li> <li>• - identifica, descrie și proiecta algoritmi specifici domeniului ingineriei autovehiculelor prin rezolvarea de aplicații</li> <li>• - testa și implementa programe( software) în domeniul ingineriei autovehiculelor conform algoritmilor realizați</li> <li>• - recunoaște componentele hardware ale unui sistem digital</li> <li>• - descrie cum poate folosi resursa informațională (motoare de căutare, reguli de bază pentru interogare, aplicațiile de editare text, imagine, video) în aplicații software din domeniul ingineriei autovehiculelor,</li> <li>• utiliza resursele de comunicare și software de specialitate pentru soluționarea problemelor (ex: utilitarul CodeBlocs), utilizând un limbaj de programare pentru scrierea programelor (ex: limbajul C – cu redactare în limba engleza)</li> <li>• - dezvoltă aplicații care citesc date din procese specifice autovehiculelor rutiere, le afișează, le procesează, demonstrând că înțelege importanța utilizării responsabile a informației și integrarea IT-ului în problemele reale</li> <li>• utilizează corect echipamentele digitale, a selecta informații relevante, redacta și salva conținut adaptat scopului, respectând totodată regulile de securitate și protecție a datelor personale</li> <li>•</li> </ul> |   |  |                             |

Data completării

10.09.2025

Titular de curs  
(semnătura)

Titular activități aplicative  
(semnătura)

Director de departament

Data avizării în Consiliul Facultății<sup>18</sup>

Decan

(semnătura)

17.09.2025

(semnătura)