



FIȘA DISCIPLINEI

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea VALAHIA din Târgoviște
1.2 Facultatea/Departamentul	Inginerie Electrică, Electronică și Tehnologia Informației
1.3 Departamentul	Electronică, Telecomunicații și Inginerie Energetică
1.4 Domeniul de studii	Inginerie Electronică, Telecomunicații și Tehnologii Informaționale
1.5 Ciclul de studii	Licență
1.6 Programul de studii/Calificarea	Tehnologii și Sisteme de Telecomunicații

2. Date despre disciplină

2.1 Denumirea disciplinei	Programarea calculatoarelor și limbaje de programare						
2.2 Titularul activităților de curs	Conf. dr. ing. Ana-Maria SUDUC						
2.3 Titularul activităților de seminar	Asist. drd. ing. Marius Adrian PĂUN						
2.4 Anul de studiu	I	2.5 Semestrul	I	2.6 Tipul de evaluare	E	2.7 Regimul disciplinei	B F

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1 Număr de ore pe săptămână	4	din care: 3.2 curs	2	3.3 seminar/laborator	2
3.4 Total ore din planul de învățământ	56	din care: 3.5 curs	28	3.6 seminar/laborator	28
Distribuția fondului de timp					Ore
Studiul după manual, suport de curs, bibliografie și notițe					15
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					5
Tutoriat					5
Examinări					4
Alte activități					-
3.7 Total ore studiu individual					44
3.9 Total ore pe semestru					100
3.10 Numărul de credite					4

4. Precondiții (acolo unde este cazul)

4.1 de curriculum	-
4.2 de competențe	-

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului	Sala cu videoproiector / tablă interactivă (pentru activități față în față) / Microsoft Teams și Moodle (pentru activități online)
5.2 de desfășurare a seminarului/laboratorului	Sala cu PC-uri (desktop sau laptop) (pentru activități față în față) / Microsoft Teams și Moodle (pentru activități online), CodeBlocks

6. Obiectivele disciplinei (reieșind din grila competențelor specifice acumulate)

6.1 Obiectivul general al disciplinei	Cunoașterea principiilor programării procedurale și formarea abilităților de programare într-un limbaj de nivel înalt
6.2 Obiectivele specifice	Folosirea conceptelor de bază din limbajul de programare C++ Formarea deprinderilor de utilizare a unui mediu de programare avansat Crearea de programe ce conțin instrucțiuni de control / tablouri / funcții / fișiere

7. Rezultatele învățării

7.1 Cunoștințe (<i>Rezultatul asimilării de informații prin învățare. Cunoștințele reprezintă ansamblul de fapte, principii, teorii și practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice și/sau faptice</i>)
<ul style="list-style-type: none"> - Absolventul cunoaște conceptele fundamentale ale programării procedurale, inclusiv tipuri de date, operatori, structuri decizionale și repetitive, funcții, tablouri, pointeri și referințe, utilizate în limbajul C++. - Absolventul înțelege principiile de organizare și structurare a programelor în C++, precum și modul de gestionare a memoriei și a fluxurilor de date. - Absolventul cunoaște mecanismele de intrare/ieșire și lucrul cu fișiere, necesare pentru realizarea aplicațiilor simple de procesare a datelor
7.2 Aptitudini (<i>Capacitatea de a aplica cunoștințe și de a utiliza know-how pentru a duce la îndeplinire sarcini și a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive și creative) sau practice (implicând dexteritate manuală și utilizarea de metode, materiale, unelte și instrumente))</i>)
<ul style="list-style-type: none"> - Absolventul este capabil să aplice conceptele de programare procedurală pentru a dezvolta programe C++ corecte și eficiente care rezolvă probleme de complexitate redusă și medie. - Absolventul este capabil să utilizeze structuri de control, funcții, tablouri și pointeri pentru implementarea algoritmilor specifici, demonstrând gândire logică și capacitate de analiză. - Absolventul este capabil să dezvolte și testeze aplicații C++ care utilizează fișiere pentru stocarea și prelucrarea datelor, folosind un mediu de dezvoltare adecvat.
7.3 Responsabilitate și autonomie (<i>Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale</i>)
<ul style="list-style-type: none"> - Absolventul are capacitatea de a lucra autonom în elaborarea, compilarea și depanarea programelor C++, respectând cerințele specificate. - Absolventul își asumă responsabilitatea pentru corectitudinea și claritatea codului scris, aplicând bune practici de programare și documentare. - Absolventul are capacitatea de autoevaluare și îmbunătățire continuă, prin analizarea erorilor și optimizarea soluțiilor implementate.

8. Conținuturi

8.1 Curs	Metode de predare	Observații
Introducere. Limbaje de programare – generalități. Paradigme de programare	problematizarea, studiul de caz, demonstrația	1h
Limbaje și translațoare. Etapele compilării și execuției. Etapele dezvoltării unui program.	problematizarea, studiul de caz, demonstrația	1h
Introducere în C++. Structura generală a unui program. Tipuri de date	problematizarea, studiul de caz, demonstrația	1h
Operatori în C++	problematizarea, studiul de caz, demonstrația	2h
Instrucțiuni de control	problematizarea, studiul de caz, demonstrația	4h
Tablouri și șiruri de caractere	problematizarea, studiul de caz, demonstrația	5h
Pointeri și referințe	problematizarea, studiul de caz, demonstrația	2h
Funcții	problematizarea, studiul de caz, demonstrația	4h
Structuri de date	problematizarea, studiul de caz, demonstrația	2h
Operații cu fișiere	problematizarea, studiul de caz, demonstrația	4h
Recapitulare	problematizarea	2h
Bibliografie		
1. Suduc, A.M., Note de curs, 2025, <i>Programarea calculatoarelor și limbaje de programare</i> , disponibile pe platforma http://moodle.valahia.ro/		
2. Reed, Mark, C++: 2 books in 1 - The Ultimate Beginners Guide to Master C++ Programming Quickly with No Prior Experience, 2023		
3. Stroustrup B., Programming. Principles and Practices Using C++, Pearson Education, 2023		
4. Torsten T. Will, C++. The Comprehensive Guide, Rheinwerk Computing, 2025		
8.2 Seminar/laborator	Metode de predare	Observații
Elemente de baza ale limbajului C++	problematizarea, studiul de caz	2h
Tipuri de date	problematizarea, studiul de caz	2h
Variabile. Operatori.	problematizarea, studiul de caz	2h
Operatori speciali. Pointeri.	problematizarea, studiul de caz	2h

Instrucțiuni de control – de selecție și de iterație. Instrucțiuni de salt și instrucțiuni etichetă	problematizarea, studiul de caz	4h
Tablouri în C++	problematizarea, studiul de caz	4h
Șiruri de caractere	problematizarea, studiul de caz	2h
Funcții în C++	problematizarea, studiul de caz	4h
Fișiere în C++	problematizarea, studiul de caz	4h
Recapitulare		2h
Bibliografie		
1. Suduc, A.M., <i>Lucrări de laborator, 2025, Programarea calculatoarelor și limbaje de programare</i> , disponibile pe platforma http://moodle.valahia.ro/		
2. Reed, Mark, <i>C++: 2 books in 1 - The Ultimate Beginners Guide to Master C++ Programming Quickly with No Prior Experience</i> , 2023		
3. Stroustrup B., <i>Programming. Principles and Practices Using C++</i> , Pearson Education, 2023		
4. Torsten T. Will, <i>C++. The Comprehensive Guide</i> , Rheinwerk Computing, 2025		

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

Întreprinderi și firme de profil din zonele și județele limitrofe orașului Târgoviște.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	- cunoașterea și aplicarea principiilor programării procedurale - cunoașterea și utilizarea corectă a tipurilor de date, a instrucțiunilor de control, a lucrului cu tablouri, șiruri, funcții și fișiere în C++	- Examinare scrisă sau sub forma unui test pe Moodle privitoare la însușirea cunoștințelor teoretice și aplicative	60%
10.5 Seminar/laborator	- utilizarea corectă a mediului de programare - elaborarea și depanarea corectă a programelor-aplicații de laborator	- examinare pe parcurs, 2 lucrări de verificare (teste pe Moodle)	40%
10.6	Standard minim de performanță		
Pentru promovarea disciplinei, studentul trebuie să îndeplinească cumulativ următoarele condiții:			
<ul style="list-style-type: none"> - obține minimum 50% din punctajul total cumulat și minimum 50% din punctajul aferent fiecărei componente de evaluare (examen final și activitate de laborator); - demonstrează cunoștințe fundamentale (RI – Cunoștințe) privind conceptele programării procedurale, tipurile de date, structurile de control, funcțiile și lucrul cu fișiere în limbajul C++; - aplică aptitudini specifice (RI – Aptitudini) pentru dezvoltarea de programe corecte și eficiente, utilizând algoritmi și structuri de date adecvate; - utilizează metode și instrumente software (RI – Aptitudini) pentru implementarea, testarea și depanarea aplicațiilor în medii de programare dedicate; - manifestă responsabilitate și autonomie (RI – Responsabilitate și autonomie) în realizarea lucrărilor de laborator, respectarea cerințelor și organizarea procesului de dezvoltare a programelor; - argumentează soluțiile implementate, demonstrând capacitate de analiză, gândire logică și optimizare a codului. 			
Standardul minim de performanță validează atingerea rezultatelor învățării definite la punctul 7 (7.1–7.3).			

Fișa disciplinei corespunde planului de învățământ care se aplică pentru anul I începând cu anul universitar 2025-2026.

Data completării
26.09.2025

Titularul de curs
Conf. univ. dr. ing. Ana-Maria SUDUC

Titularul de aplicații
As. univ. drd. ing. Marius Adrian PĂUN

Data avizării în
departament
29.09.2025

Director de departament
Conf. univ. dr. ing. Dan Constantin PUCHIANU

Data avizării în
Consiliul Facultății
30.09.2025

Decan
Conf. univ. dr. ing. Nicoleta ANGELESCU