



## FIȘA DISCIPLINEI

### 1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Valahia din Târgoviște
1.2 Facultatea	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	Tehnologii și Sisteme de Telecomunicații

### 2. Date despre disciplină

2.1 Denumirea disciplinei	<b>Programare obiect-orientată</b>						
2.2 Titularul activităților de curs	Conf. dr. ing. Mihai BÎZOI						
2.3 Titularul activităților de seminar	Conf. dr. ing. Mihai BÎZOI						
2.4 Anul de studiu	II	2.5 Semestrul	I	2.6 Tipul de evaluare	C	2.7 Regimul disciplinei	B-D

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

3.1 Număr de ore pe săptămână	3	din care: 3.2 curs	2	3.3 seminar/laborator	1
3.4 Total ore din planul de învățământ	42	din care: 3.5 curs	28	3.6 seminar/laborator	14
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					30
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					20
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					0
Tutoriat					2
Examinări					2
Alte activități .....					4
3.7 Total ore studiu individual					58
3.9 Total ore pe semestru					100
3.10 Numărul de credite					4

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

4.1 de curriculum	<ul style="list-style-type: none"><li>• Programarea calculatoarelor și limbaje de programare</li><li>• Structuri de date și algoritmi</li></ul>
4.2 de competențe	Descrierea funcționării unui sistem de calcul, a principiilor de bază ale arhitecturii microprocesoarelor și microcontrolerelor de uz general, a principiilor generale ale programării structurate.

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

5.1 de desfășurare a cursului	Sală de curs cu videoproiector și calculator sau tablă interactivă. În funcție de situația epidemiologică, cursurile se vor desfășura online, în conformitate cu procedura aplicată, pe platformele Moodle și Teams.
5.2 de desfășurare a seminarului/laboratorului	Sală de laborator cu calculatoare și software specific (mediul de Programare C++ CodeBlocks).

În funcție de situația epidemiologică, laboratoarele se vor desfășura online, în conformitate cu procedura aplicată, pe platformele Moodle și Teams.

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

6.1 Obiectivul general al disciplinei	Formarea conceptelor de baza din programarea orientată pe obiecte. Însușirea deprinderilor de a utiliza instrumentele și tehnologiile de proiectare și realizare a aplicațiilor orientate pe obiecte.
6.2 Obiectivele specifice	<ul style="list-style-type: none"> <li>• Înțelegerea principiilor programării orientată pe obiecte.</li> <li>• Cunoașterea sintaxei și a modului de supraîncărcare a operatorilor.</li> <li>• Înțelegerea modului de realizare a aplicațiilor orientate pe obiecte într-un limbaj hibrid.</li> <li>• Proiectarea aplicațiilor orientate pe obiecte cu interfață grafică.</li> </ul>

## 7. Rezultatele învățării

<p>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>)</p> <p>RIC1. Explică paradigmele de programare și conceptele fundamentale ale programării orientate pe obiecte, cu accent pe limbajul C++ și diferențele față de alte limbaje orientate pe obiecte.</p> <p>RIC2. Descrie mecanismele de definire și utilizare a claselor și obiectelor, inclusiv mesaje, metode, constructori, destructori și principiile încapsulării.</p> <p>RIC3. Explică moștenirea simplă și multiplă, polimorfismul, funcțiile virtuale și clasele abstracte, precum și implicațiile acestora în proiectarea aplicațiilor.</p> <p>RIC4. Prezintă regulile de supraîncărcare a operatorilor, utilizarea funcțiilor și claselor prieten și conceptele de programare generică prin template-uri.</p> <p>RIC5. Descrie mecanismele de intrare-ieșire în C++, principalele componente ale bibliotecii standard (STL) și rolul diagramelor de clase UML în dezvoltarea aplicațiilor orientate pe obiecte.</p>
<p>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>)</p> <p>RIA1. Proiectează și implementează aplicații orientate pe obiecte de complexitate redusă, utilizând clase, moștenire și polimorfism în C++.</p> <p>RIA2. Utilizează corect constructori, destructori, supraîncărcarea operatorilor și mecanismele de copiere și atribuire.</p> <p>RIA3. Aplică funcții și clase template, precum și containere și algoritmi din biblioteca standard C++.</p> <p>RIA4. Implementează mecanisme de intrare-ieșire pentru tipuri predefinite și definite de utilizator.</p> <p>RIA5. Modelează aplicații simple utilizând diagrame de clase UML și le transpune în cod C++.</p> <p>RIA6. Dezvoltă și evaluează aplicații orientate pe obiecte simple, cu interfață grafică, în cadrul activităților de laborator.</p>
<p>7.3 Responsabilitate și autonomie (<i>Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale</i>)</p> <p>RIr1. Rezolvă autonom sarcini de programare orientată pe obiecte, respectând cerințele tehnice și bunele practici de dezvoltare software.</p> <p>RIr2. Utilizează responsabil resursele software, documentația tehnică și bibliotecile asociate limbajului C++.</p> <p>RIr3. Colaborează eficient în activitățile de laborator și respectă normele etice și academice în realizarea aplicațiilor software.</p> <p>RIr4. Manifestă interes pentru perfecționarea continuă a competențelor în domeniul programării orientate pe obiecte.</p>

## 8. Conținuturi

8.1 Curs	Metode de predare	Observații
C1. Introducere (Paradigme de programare; Evoluția limbajelor de programare orientate obiect; Obiect, mesaje, metode; Comparatie limbaj pur versus hibrid)	Expunerea, prezentarea, conversația	2 ore
C2. Clase și instanțe (Mesaje între obiecte; Clase și instanțe; Evaluarea metodelor; Principiile programării orientate pe obiecte)	Expunerea, prezentarea, conversația	2 ore
C3. Constructori și destructori (Constructorii; Caracteristici constructori; Tipuri de constructori; Destructori; Caracteristici destructori)	Expunerea, prezentarea, conversația	2 ore
C4. Moștenirea în limbajul C++ (Avantajele moștenirii; Sintaxa definirii claselor derivate; Accesul asupra membrilor moșteniți; Tipuri de moștenire; Constructori / destructori în procesul de moștenire)	Expunerea, prezentarea, conversația	2 ore
C5. Derivarea multiplă (Derivarea multiplă. Problema diamantului; Clase virtuale; Constructorul de copiere în procesul de moștenire; Supraîncărcarea operatorului de atribuire)	Expunerea, prezentarea, conversația	2 ore
C6. Polimorfismul (Ce este polimorfismul; Tipuri de polimorfism; Funcții virtuale; Clase abstracte)	Expunerea, prezentarea, conversația	2 ore

C7. Supraîncărcarea operatorilor. Funcții și clase prieten (Supraîncărcarea operatorilor: sintaxă și reguli; Funcții prieten: globale și membre ale altor clase; Clase prieten)	Expunerea, prezentarea, conversația	2 ore
C8. Funcții și clase template (șablon) (Funcții template: sintaxă, exemple; Clase template: sintaxă, exemple;)	Expunerea, prezentarea, conversația	2 ore
C9. Sistemul de intrare-ieșire în C++ (Sistemul de intrare/ieșire în C++; Funcții de I/O pentru tipurile predefinite; Funcții de I/O pentru tipuri definite de utilizator; Starea streamurilor; Formatarea I/O)	Expunerea, prezentarea, conversația	2 ore
C10. Diagrame de clase UML (Diagrame UML; Reprezentarea claselor; Specificarea atributelor și metodelor; Relațiile dintre clase; Reutilizarea codului)	Expunerea, prezentarea, conversația	2 ore
C11. Biblioteca standard (Containere; Reprezentare; Operații; Tipuri de containere)	Expunerea, prezentarea, conversația	2 ore
C12. Biblioteca standard (Algoritmi; Algoritmi care modifică sau nu secvența; Sortare și căutare; Minim și maxim)	Expunerea, prezentarea, conversația	2 ore
C13. Dezvoltarea aplicațiilor orientate obiect (Evoluția proceselor de dezvoltare software; Modelarea folosind cazuri de utilizare; De la cazuri de utilizare la clase; Scrierea codului)	Expunerea, prezentarea, conversația	2 ore
C14. Recapitulare (subiecte posibile la verificarea scrisă)	Expunerea, prezentarea, conversația	2 ore

#### Bibliografie

1. Mihai Bizoi, Programare obiect-orientată – notițe de curs, <https://moodle.valahia.ro>;
2. Bjarne Stroustrup, *A Tour of C++ (Third Edition)*, Pearson Education, 2023.
3. Bjarne Stroustrup, *Programming: Principles and Practice Using C++ (Third Edition)*, Pearson Education, 2024.
4. Marc Gregoire, *Professional C++ Sixth Edition*, John Wiley & Sons, 2024.
5. John Lakos, Vittorio Romeo, Rostislav Khlebnikov, Alisdair Meredith, *Embracing Modern C++ Safely*, Pearson Education, 2022.
6. J. Guy Davidson & Kate Gregory, *Beautiful C++*, Pearson Education, 2022.
7. Federico Busato (Nvidia), *Modern CPP Programming*, 2024.

8.2 Seminar/laborator	Metode de predare	Observații
L1. Prezentarea noțiunilor de clasă și obiect.	Expunerea, demonstrația, lucrări practice	2 ore
L2. Realizarea interfețelor grafice. Instalarea și configurarea wxWidgets.	Expunerea, demonstrația, lucrări practice	2 ore
L3. Constructori și destructori.	Expunerea, demonstrația, lucrări practice	2 ore
L4. Supraîncărcarea operatorilor. Funcții prieten.	Expunerea, demonstrația, lucrări practice	2 ore
L5. Derivarea (moștenirea) claselor.	Expunerea, demonstrația, lucrări practice	2 ore
L6. Polimorfism. Supraîncărcarea operatorilor.	Expunerea, demonstrația, lucrări practice	2 ore
L7. Evaluare practică.		2 ore

#### Bibliografie

1. Bizoi M., Programare obiect-orientată – lucrări de laborator, <https://moodle.valahia.ro>
2. Bjarne Stroustrup, *A Tour of C++ (Third Edition)*, Pearson Education, 2023.
3. Bjarne Stroustrup, *Programming: Principles and Practice Using C++ (Third Edition)*, Pearson Education, 2024.
4. Marc Gregoire, *Professional C++ Sixth Edition*, John Wiley & Sons, 2024.

## 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	Testarea noțiunilor teoretice fundamentale	Examinare scrisă (lucrare scrisă sau test online)	50%
10.5 Seminar/laborator	Testarea noțiunilor teoretice fundamentale	Test grilă pe Moodle	20%
	Realizarea de aplicații funcționale	Verificare practică (în laborator sau pe MS Teams)	30%
10.6	Standard minim de performanță		

Pentru promovarea disciplinei, studentul trebuie să îndeplinească cumulativ următoarele condiții:

- obține minimum 50% din punctajul total cumulat și minimum 50% din punctajul aferent fiecărei componente de evaluare (examen scris, testare pe parcurs și evaluare practică de laborator);
- demonstrează cunoștințe fundamentale (RI – Cunoștințe) privind conceptele programării orientate pe obiecte, clasele și obiectele, moștenirea, polimorfismul și utilizarea bibliotecii standard C++;
- aplică aptitudini specifice (RI – Aptitudini) pentru proiectarea și implementarea aplicațiilor orientate pe obiecte utilizând limbajul C++;
- utilizează metode și instrumente (RI – Aptitudini) pentru dezvoltarea, testarea și optimizarea aplicațiilor software, inclusiv utilizarea structurilor STL și a mecanismelor de intrare-ieșire;
- manifestă responsabilitate și autonomie (RI – Responsabilitate și autonomie) în realizarea aplicațiilor, respectarea cerințelor și utilizarea corectă a resurselor software;
- argumentează soluțiile implementate, demonstrând capacitate de analiză, proiectare și dezvoltare a aplicațiilor orientate pe obiecte.

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 2024-2025.

Data completării  
26.09.2025

Titularul de curs  
Conf. dr. ing. Mihai BÎZOI

Titularul de aplicații  
Conf. dr. ing. Mihai BÎZOI

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