

FIȘA DISCIPLINEI
Anul universitar 2024-2025

Decan,
Conf.dr.ing. Gelu Ianus

1. Date despre program

1.1 Instituția de învățământ superior	Universitatea Tehnică „Gheorghe Asachi” din Iași
1.2 Facultatea	MECANICA
1.3 Departamentul	Inginerie Mecanica, Mecatronica si Robotica
1.4 Domeniul de studii	MECATRONICA SI ROBOTICA
1.5 Ciclul de studii ¹	MASTER
1.6 Programul de studii	MECATRONICA AVANSATA

2. Date despre disciplină

2.1.1 Denumirea disciplinei	PROGRAMAREA ORIENTATA PE OBIECTE <i>Object Oriented Programming</i>						
2.1.2 Codul disciplinei	MA.IA.203						
2.2 Titularul activităților de curs	Conf.dr.ing. Vasile-Ciprian STAMATE						
2.3 Titularul activităților de aplicații	Asist.dr.ing. Cezara Mariuca Gavril						
2.4 Anul de studii ²	2	2.5 Semestrul ³	3	2.6 Tipul de evaluare ⁴	C	2.7 Tipul disciplinei ⁵	DI.DA.

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

3.1 Număr de ore pe săptămână										3	din care 3.2 curs	1	3.3a sem.	-	3.3b laborator	2	3.3c proiect	-
3.4 Total ore din planul de învățământ ⁶										42	din care 3.5 curs	14	3.6a sem.	-	3.6b laborator	28	3.6c proiect	-
Distribuția fondului de timp ⁷																	Nr. 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/proiecte, teme, referate și portofolii																	10	
Tutoriat ⁸																	8	
Examinări ⁹																	10	
Alte activități:																		
3.7 Total ore studiu individual ¹⁰										58								
3.8 Total ore pe semestru ¹¹										100								
3.9 Numărul de credite										4								

4. Precondiții (acolo unde este cazul)

4.1 de curriculum ¹²	• Programarea calculatoarelor si limbaje de programare, Software pt programarea microcontrolerelor
4.2 de competențe	• Implementarea algoritmilor in C++

5. Condiții (acolo unde este cazul)

5.1 de desfășurare a cursului ¹³	• Sala cu tabla si videoproiector
5.2 de desfășurare a seminarului / laboratorului / proiectului ¹⁴	• Laborator de informatica

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

6.1 Obiectivul general al disciplinei	• Familiarizarea studenților cu tehnici de bază specifice programării orientate obiect
6.2 Obiective specifice	• Implementare in C++ a tehnicilor POO

7. Rezultatele învățării

Cunoștințe	- cunoaște conceptele fundamentale ale programării orientate pe obiecte (POO): clase, obiecte, moștenire, polimorfism, încapsulare, abstractizare; - înțelege sintaxa și semantica limbajului C++ în contextul paradigmei orientate pe obiecte; - cunoaște structura ierarhică a claselor, tipurile de relații dintre clase (moștenire, prietenie, interfețe); - înțelege mecanismele de implementare a funcțiilor virtuale, clase abstracte și metode suprascrise; - este familiarizat cu modelarea și proiectarea obiectuală a sistemelor software, inclusiv în domeniul mecatronicii.
------------	--

Aptitudini	• implementează în C++ aplicații folosind tehnici avansate de POO (pointeri, referințe, clase abstracte, interfețe); • modelează relațiile dintre obiecte folosind moștenire, polimorfism și compoziție; • testează, validează și documentează codul sursă dezvoltat în cadrul laboratorului; • analizează probleme ingineresti și propune soluții software structurate în stil orientat pe obiecte; • utilizează eficient resursele de programare pentru dezvoltarea de componente reutilizabile în C++ pentru sisteme mecatronice; • adaptează exemple practice la contexte specifice industriei ingineresti sau academice.
Responsabilitate și autonomie	• își asumă responsabilitatea pentru organizarea codului în mod coerent, clar și reutilizabil; • manifestă autonomie în procesul de învățare și aplicare a noțiunilor avansate de programare orientată pe obiecte; • contribuie activ la dezvoltarea și evaluarea proiectelor software în cadrul activităților de laborator; • colaborează eficient cu colegii în rezolvarea temelor și proiectelor de echipă; • respectă bunele practici în scrierea codului (naming, structurare, comentarii, modularizare); • se implică în documentarea și autoevaluarea performanței în vederea îmbunătățirii continue a competențelor de programare.

8. Conținuturi

8.1 Curs ¹⁵	Metode de predare ¹⁶	Observații
Curs 1: Limbajul C ca subset al limbajului C++ Curs 2: Introducere în POO.Principiile POO. Clase si obiecte Curs 3: Introducere în POO. Obiecte, referinte si pointeri la obiecte Curs 4: Introducere în POO.Modelare si proiectare OO Curs 5:Mostenirea în POO. Notiunea de mostenire și descrierea ierarhiilor de clase Curs 6: Mostenirea în POO. Funcții virtuale și implementarea polimorfismului Curs 7: Mostenirea în POO. Clase abstracte, clase friend interfețe, clase parametrizate	Prelegere clasica.Expunere cu videoprojector. Conversatii	14 ore
Bibliografie curs: 1. Herbert Schildt: C++ Manual Complet, Bucuresti, Ed. Teora 2000 2. Bruce Eckel, Thinking in C++, Prentice Hall, 2000 3. Kris Jamsa si Lars Klander, Totul despre C si C++ Manualul fundamental de programare în C si C++, Ed. Teora, (traducere 2007); 4. Stroustrup, B.,1991, The C++ Programming Language - 2nd edition, Addison-Wesley Publishing Company		
8.2b Laborator	Metode de predare ¹⁷	Observații
1 Limbajul C. Recapitulare 2 Principiile POO. Clase si obiecte 3 Obiecte, referinte si pointeri la obiecte 4 Prezentarea limbajului C++, cu accent pe reprezentarea claselor și obiectelor 5 Modelare si proiectare OO 6 Notiunea de mostenire și descrierea ierarhiilor de clase 7 Funcții virtuale și implementarea polimorfismului 8 Clase abstracte, clase friend interfețe, clase parametrizate	Implementare/testare cod sursă, problematizarea	28 ore
Bibliografie aplicații (seminar / laborator / proiect): 1. Stroustrup, B.,1991, The C++ Programming Language - 2nd edition, Addison-Wesley Publishing Company 2. Buzzi-Ferraris, G., 1993, Building Numerical Libraries the Object-Oriented Way, Addison-Wesley Publishing Company 3. Herbert Schildt: C++ Manual Complet, Bucuresti, Ed. Teora 2000 4. Thinking in C++, Bruce Eckel, Prentice Hall, 2000 5. Kris Jamsa si Lars Klander, Totul despre C si C++ Manualul fundamental de programare în C si C++, Ed. Teora, (traducere 2007)		

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului¹⁸

- Cursul și laboratorul vor furniza studenților pregătirea necesară în ceea ce privește domeniul mecatronicii .

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
----------------	---------------------------	-------------------------	------------------------------

10.4aExamen / Colocviu	• Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea)	Evaluare finală: test pe calculator	100% (minim 5)	100% (minim 5)
10.5 Standard minim de performanță ¹⁹				

Data completării,

10.09.2024

Semnătura titularului de curs,

Conf.dr.ing. Vasile-Ciprian STAMATE

Semnătura titularului de aplicații,

Asist.dr.ing. Cezara Mariuca Gavril

Data avizării în departament,

12.09.2024

Director departament,

Prof.dr.ing. Ioan DOROFTEI

¹Licență / Master

² 1-4 pentru licență, 1-2 pentru master

³ 1-8 pentru licență, 1-3 pentru master

⁴Examen, colocviu sau VP A/R – din planul de învățământ

⁵DF - disciplină fundamentală, DID - disciplină în domeniu, DS – disciplină de specialitate sau DC - disciplină complementară - din planul de învățământ

⁶ Este egal cu 14 săptămâni x numărul de ore de la punctul 3.1 (similar pentru 3.5, 3.6abc)

⁷ Liniile de mai jos se referă la studiul individual; totalul se completează la punctul 3.7.

⁸Între 7 și 14 ore

⁹Între 2 și 6 ore

¹⁰ Suma valorilor de pe liniile anterioare, care se referă la studiul individual.

¹¹ Suma dintre numărul de ore de activitate didactică directă (3.4) și numărul de ore de studiu individual (3.7); trebuie să fie egală cu numărul de credite alocate disciplinei (punctul 3.9) x 24 de ore pe credit.

¹²Se menționează disciplinele obligatoriu a fi promovate anterior sau echivalente

¹³Tablă, videoproiector, flipchart, materiale didactice specifice etc.

¹⁴Tehnică de calcul, pachete software, standuri experimentale, etc.

¹⁵Titluri de capitole și paragrafe

¹⁶Expunere, prelegere, prezentare la tablă a problematicei studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

¹⁷Demonstrație practică, exercițiu, experiment

¹⁸Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

¹⁹ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.