

FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Inginerie Electrică și Știința Calculatoarelor
Departamentul	Calculatoare
Domeniul de studii	Calculatoare și Tehnologia Informației
Ciclul de studii	Licență, Dual
Programul de studii	Calculatoare

2. Date despre disciplină

Denumirea disciplinei	PROGRAMARE ORIENTATĂ PE OBIECTE				
Anul de studiu	II	Semestrul	III	Tipul de evaluare	E
Regimul disciplinei	Categoría formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				DF
	Categoría de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă				DOB

3. Timpul total estimat (ore alocate activităților didactice)

I a) Număr de ore pe săptămână	Total genera l	5	Curs	3	Seminar		Laborator IIS		Proiect IIS		Practică IIS	
I b) Totalul de ore pe semestru din planul de învățământ		70	Curs	42	Seminar		Laborator IM	2	Proiect IM		Practică IM	
							Laborator	28	Proiect		Practică	

(IIS – instituție de învățământ superior; IM – învățare prin muncă)

Distribuția fondului de timp pe semestru	Ore IIS	Ore IM
II.a) Studiu individual	5	47
II.b) Tutoriat (pentru ID)		
III. Examinări	3	
IV. Alte activități (precizați):		

Total ore studiu individual (II.a+II.b+III)	Ore IIS	8	Ore IM	47
Total ore pe semestru (Ib+II+III+IV)	Ore IIS	50	Ore IM	75
Numărul de credite	Credite IIS	2	Credite IM	3

4. Competențe specifice acumulate

Competențe profesionale	CP4-definește arhitectura software; CP5-creează softuri; CP6-definește cerințe tehnice; CP7-dezvolta prototipul pentru software; CP16-utilizează biblioteci de software;
Competențe transversale	

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul/absolventul descrie, identifică, sumarizează concepte și metode elementare privitoare la limbaje de programare, medii de	Studentul/absolventul alege și explică concepte proprii specifice proiectării algoritmilor, programării	Studentul/absolventul are o comportare onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei.

programare, tehnici de programare, baze de date, inteligență artificială și inginerie software și modul lor de aplicare în probleme concrete	orientate pe obiecte, programării logice și funcționale. Studentul/absolventul specifică cerințe, analizează, elaborează, dezvoltă și testează programe în limbaje de programare de uz general (C, etc.) și /sau obiect-orientate (C++, Java, etc.), aplicând elementele specifice ingineriei software.	
--	--	--

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

Obiectivul general al disciplinei	Însușirea elementelor fundamentale privind paradigmele de programare orientata pe obiecte, respectiv asimilarea fundamentelor limbajului C++
	O1. Însușirea conceptelor și paradigmelor programării orientate spre obiecte; O2. Studiarea limbajului de programare C++; O3. Însușirea tehnicilor de programare orientată spre obiecte; O4. Continuarea prezentării tipurilor de date fundamentale si studiul STL, biblioteca standard C++; O5. Pregătirea pentru studiul algoritmilor fundamentali si introducerea in noul standard C++0x.

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1. Programarea Orientata pe Obiecte 1.1. Fundamentele POO 1.2. Conceptele ce definesc tehnologiile orientării obiect	3	expunerea, prelegerea-dezbateri, demonstrația	
2. Clase. Membrii unei clase 2.1. Autoreferința. Tipul de data referința 2.2. Obiecte standard de interfață (stream-uri) 2.3. Constructori și destructori 2.4. Constructorul de copiere 2.5. Membri statici ai unei clase 2.6. Tablouri de obiecte 2.7. Clase cu membri obiecte 2.8. Pointeri spre membri unei clase 2.9. Funcții și clase prietene ale unei clase.	9	expunerea, prelegerea-dezbateri, demonstrația	
3. Supradefinirea operatorilor 3.1. Operatorii unari și binari 3.2. Operatori speciali (new, delete, ->, apel funcție, etc.) 3.3. Operatori și constructori de conversie de tip	6	expunerea, prelegerea-dezbateri, demonstrația	
4. Conceptele de Moștenire și Generalizare 4.1. Moștenirea simplă 4.2. Moștenirea multiplă 4.3. Clase și funcții virtuale	6	expunerea, prelegerea-dezbateri, demonstrația	
5. Polimorfismul 5.1. Funcții virtuale 5.2. Funcții pur virtuale	6	expunerea, prelegerea-dezbateri, demonstrația	
6. Mecanismul Template 6.1. Funcții parametrizate (template) 6.2. Clase parametrizate (template) 6.3. Mecanismul Template și moștenirea	3	expunerea, prelegerea-dezbateri, demonstrația	
7. Mecanismul de tratare a excepțiilor în C++ 7.1. Erori și excepții 7.2. Tratarea erorilor 7.3. Excepții în C++. Mecanismul <i>throw, try...catch</i>	3	expunerea, prelegerea-dezbateri, demonstrația	
8. Elemente specifice limbajului C++ 8.1. Spații de nume (namespace)	3	expunerea, prelegerea-dezbateri, demonstrația	

8.2. Conversii de tip (type-casting)			
9. Standard Template Library (STL) 9.1. Containere STL 9.2. Algoritmi STL	2	expunerea, prelegerea-dezbatare, demonstrația	
10. Evoluția limbajului: C++0x	1	expunerea, prelegerea-dezbatare, demonstrația	
Bibliografie minimală recomandată			
1. B.Stroustrup, The A Tour of C++ (2nd edition), Addison-Wesley. ISBN 978-0-13-499783-4. July 2018; 2. B.Stroustrup, The C++ Programming Language, 4th Edition, Addison-Wesley, ISBN: 0321563840, May 2013; 3. Björn Andrist, V. Sehr, C++ High Performance (2nd edition), Packt Publishing, ISBN 978-1-83921-654-1, 2020; 4. Remus Prodan – pagina web a cursului: http://dev.fiesc.usv.ro/remus 5. C++ Primer, Fifth edition, Addison Wesley Publ., 2013. 6. B.Stroustrup, C++, Editura Teora, ISBN: 973-20-0894-6 , 2003 7. Ioan Salomie, Tehnici orientate pe obiecte, Ed. Microinformatica Cluj, 1995. 8. M. Preda, A. Mirea, D. Preda, C.Teodorescu. Introducere in programarea orientata-obiect. Concepte fundamentale din perspectiva ingineriei software, Editura Polirom. 2010 9. Șt.Gh. Pentiuc - Structuri de date și algoritmi fundamentali. Curs, Universitatea "Ștefan cel Mare" Suceava, 1993. 10. http://www.cplusplus.com/			

Aplicații IM (laborator)	Nr. ore	Metode de predare	Observații
1. Prezentarea generală a laboratorului în care se vor desfășura activitățile practice pentru disciplina "Programare Orientată pe Obiecte". Prezentarea normelor de protecția muncii. Familiarizarea cu sistemele de calcul și mediile de programare. Introducere în limbajul C++	2	expunere, verificare/testare	
2. Adăugiri aduse de C++ standardului ANSI C. Utilitarul MAKE	2	lucrări practice, experimentul	
3. Utilizarea obiectelor cin, cout și operatorul new.	2	lucrări practice, experimentul, verificare/testare	
4. Membri statici ai unei clase. Funcții și clase prietene ale unor alte clase.	2	lucrări practice, experimentul	
5. Supradefinirea operatorilor (unari și binari)	2	lucrări practice, experimentul	
6. Supradefinirea operatorilor speciali	2	lucrări practice, experimentul	
7. Stream-uri. Iteratorii.	2	lucrări practice, experimentul	
8. Moștenirea simplă. Moștenirea multiplă	4	lucrări practice, experimentul	
9. Polimorfismul	2	lucrări practice, experimentul	
10. Mecanismul template	4	lucrări practice, experimentul	
11. Tratarea excepțiilor în C++. Elemente noi în standardul C++11	4	lucrări practice, experimentul	
Temele de casă vor consta din dezvoltarea unor programe de complexitate medie. Etapă principală în dezvoltarea temei - însușirea temei de realizare; - proiectarea logică a produsului program; - programare; inserarea de comentarii în codul sursă - punerea la punct a produsului program; - testarea cu date impuse;			
Bibliografie minimală recomandată			
1. Remus Prodan – pagina web a cursului: http://dev.fiesc.usv.ro/remus 2. B.Stroustrup, The C++ Programming Language, 4th Edition, Addison-Wesley, ISBN-10: 0321563840, May 2013 3. B.Stroustrup, C++, Editura Teora, ISBN: 973-20-0894-6 , 2003 4. Ioan Salomie, Tehnici orientate pe obiecte, Ed. Microinformatica Cluj, 1995. 5. Juan Soulie, C++ Language Tutorial (Tutorial online) http://www.cplusplus.com/doc/tutorial/			

8. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
----------------	----------------------	--------------------	-------------------------

Curs	Criteriile generale de evaluare (corectitudinea cunoștințelor, coerența logică, fluența în redactare) Criterii specifice disciplinei (utilizarea corectă a metalimbajului specific domeniului limbajelor de programare)	Evaluare sumativă prin examinare scrisă (<i>proba orală și proba practică pe calculator</i>)	50
Seminar	-		
Laborator/ Lucrări practice	Criteriile generale de evaluare (corectitudinea cunoștințelor, coerența logică, fluența în redactare) Criterii specifice disciplinei (utilizarea corectă a metalimbajului specific domeniului limbajelor de programare)	Evaluare pe parcurs prin examinare scrisă (<i>prin scrierea și punerea la punct a unui program funcțional pe calculator</i>)	50
Proiect			

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație
25.09.2025	s.l. univ.dr.ing. Remus Cătălin PRODAN	as.dr.inf. Laura BILIUS

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
25.09.2025	conf. dr. ing. Ioan UNGUREAN

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
25.09.2025	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
26.09.2025	Prof. dr. ing. Laurentiu- Dan MILICI