

FIȘA DISCIPLINEI

Anul universitar 2025 - 2026

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
1.2. Facultatea	Științe
1.3. Departament	Matematică și Informatică
1.4. Domeniul de studiu	Matematică
1.5. Ciclul de studii ¹	Licență
1.6. Specializarea	Matematică și Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	PROGRAMARE ORIENTATA OBIECT	Cod	FSTI.MAI.MIL.FO.2.2 020.E-5.2
2.2. Titular activități de curs	Lector Univ. Dr. Oana-Adriana Țicleanu		
2.3. Titular activități practice	Lector Univ. Dr. Oana-Adriana Țicleanu		
2.4. An de studiu ²	3	2.5. Semestrul ³	6
2.6. Tipul de evaluare ⁴	E		
2.7. Regimul disciplinei ⁵	O	2.8. Categoria formativă a disciplinei ⁶	F

3. Timpul total estimat

3.1. Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână					
3.1.a.Curs	3.1.b. Seminar	3.1.c. Laborator	3.1.d. Proiect	3.1.e Alte	Total
2		2			4
3.2. Extinderea disciplinei în planul de învățământ – total ore din planul de învățământ					
3.2.a.Curs	3.2.b. Seminar	3.2.c. Laborator	3.2.d. Proiect	3.2.e Alte	Total ⁷
28		28			56
Distribuția fondului de timp pentru studiu individual⁸					Nr. ore
Studiul după manual, suport de curs, bibliografie și notițe					20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					15
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					25
Tutoriat ⁹					5
Examinări ¹⁰					4
3.3. Total ore alocate studiului individual¹¹ (NOSI_{sem})					69
3.4. Total ore din Planul de învățământ (NOAD_{sem})					56
3.5. Total ore pe semestru¹² (NOAD_{sem} + NOSI_{sem})					125
3.6. Nr ore / ECTS					25
3.7. Număr de credite¹³					5

4. Precondiții (acolo unde este cazul)

4.1. Discipline necesar a fi promovate anterior (de curriculum) ¹⁴	Programare procedurala
4.2. Competențe	

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului ¹⁵	Sală de curs, dotată cu tablă și videoproiector
5.2. De desfășurare a activităților practice (lab/sem/pr/alte) ¹⁶	Sală de laborator dotată cu calculatoare și videoproiector

6. Rezultate ale învățării¹⁷

Numărul de credite alocate disciplinei: 5				
Rezultatele învățării				Repartizare credite pe rezultatele învățării
Nr. crt.	Cunoștințe	Aptitudini	Responsabilitate și autonomie	
RÎ 1	Defineste clase și obiecte; explică încapsularea și nivelurile de acces (public/protected/private) și rolul lor în proiectare.	Proiectează o clasă simplă (atribute + metode), aplică încapsularea (get/set cu validare elementară) și o integrează într-un program mic.	Respectă convențiile de stil și documentează succint; verifică rezultate pe exemple de test ușoare și corectează erori de bază.	2
RÎ 2	Describe ciclul de viață al obiectelor: constructori (implicit/parametrizat/copiere/mutare, fără detalii avansate), destructor, copiere vs. atribuire.	Scrie constructori adecvați și destructor; implementează corect copierea/atribuirea pentru clase cu membri simpli; tratează cazuri de intrare invalidă.	Argumentează pe scurt alegerile (inițializare, validare) și menține claritatea codului prin refactorizări mici.	2
RÎ 3	Explică ideea de reutilizare prin moștenire și noțiunea de polimorfism (metode virtuale) la nivel introductiv; enumeră exemple uzuale.	Construiește o ierarhie minimă (bază - derivată) și demonstrează apel polimorfic prin pointer/referință la baza.	Planifică și realizează un proiect OOP (design - implementare - testare) în termenele stabilite, cu structură de proiect clară (foldere separate pentru cod și exemple, README de rulare).	1

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

7.1. Obiectivul general	Familiarizarea cu principiile programării orientate obiect;
7.2. Obiectivele specifice	Formarea deprinderilor de a proiecta aplicații utilizând principiile programării orientate obiect;

8. Conținuturi

8.1. Curs ¹⁸	Metode de predare ¹⁹	Nr. ore
Introducere în OOP. Concepte de bază.		2
Clase și obiecte.		2

Protectia datelor si functiilor membru.	Explicația, Descrierea și exemplificarea, Demonstrația, Problematizarea, Conversația euristică, Exercițiu	
Crearea si distrugerea obiectelor. Constructori. Tipuri de constructori. Destructorii.		2
Alocarea dinamica de memorie.		2
Date si functii membre statice.		2
Clase si functii friend.		2
Supraincarea operatorilor friend, membru si operatori speciali.		2
Tehnici de reutilizare a codului.		4
Mostenirea. Polimorfismul – clase si functii virtuale.		
Clase si functii template.		2
Tratarea exceptiilor in limbajul C++.		2
Operatii de I/O. Stream-uri.		2
Biblioteci Standard Template Library.		2
Sabloane de proiecte.		2
Total ore curs:		28
8.2. Activități practice (8.2.a. Seminar ²⁰ / 8.2.b. Laborator ²¹ / 8.2.c. Proiect ²² / 8.2.d. Alte act.practice ²³)	Metode de predare	Nr. ore
Introducere in Programarea Orientata pe Obiect.	Exemplificare, lucru individual, corectarea erorilor de logică a implementării, testare, expunere optimizări	2
Clasa ca tip abstract de date. Structura clasei. Componente: attribute, metode.		2
Domeniul public, private, protected. Exemple de aplicatii.		2
Implementarea aplicatiilor folosind clase, constructori si destructori.		2
Implementarea aplicatiilor folosind alocarea dinamica de memorie. Date si functii statice intr-o clasa.		2
Implementarea aplicatiilor folosind functii friend.		2
Supraincarea operatorilor.		2
Constructia de aplicatii folosind conceptul de mostenire.		4
Constructia de aplicatii folosind conceptul de polimorfism – clase si functii virtuale.		2
Implementarea de aplicatii folosind clase abstracte.		2
Implementarea de aplicatii folosind clase template.		2
Tratarea exceptiilor in OOP		2
Prezentare/Dezbatere proiecte		2
Total ore seminar/laborator		28

9. Bibliografie

9.1. Referințe bibliografice recomandate	Kamini Solanki: Object Oriented Programming Concepts Using C++, Notion Press Publisher, 2021
	Scott Meyers: Effective C++: 55 Specific Ways to Improve Your Programs and Designs, 3rd Edition, Addison-Wesley Professional Publisher, 2005
	Bruce Eckel: Thinking in C++, 2nd Edition, Prentice Hall 2000
	Bjarne Stroustrup: The C++ Programming Language, Addison-Wesley, 3rd edition, 1997
9.2. Referințe bibliografice suplimentare	Liviu Negrescu: Limbajele C si C++ pentru începatori, Vol. II, (editia XI), Editura Albastra, Cluj-Napoca, 2005

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului²⁴

Competențele dobândite în cadrul disciplinei permit absolvenților să utilizeze eficient a tehnicilor OOP în vederea implementării soluțiilor avansate din industria software. Din aceasta rezultă cultura obligatorie pentru stabilirea elementelor fundamentale, practice și analitice, necesar a fi cunoscute de un absolvent.

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare		11.3 Pondere din nota finală	Obs. ²⁵
11.4a Examen / Colocviu	• Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea)	Teste pe parcurs ²⁶ :	%	40% (minim 5)	CEF
		Teme de casă:	%		
		Alte activități ²⁷ :	%		
		Evaluare finală:	100%		
11.4b Seminar	• Frecvența/relevanța intervențiilor sau răspunsurilor	Evidența intervențiilor, portofoliu de lucrări (referate, sinteze științifice)		40% (minim 5)	N/A
11.4c Laborator	• Cunoașterea aparaturii, a modului de utilizare a instrumentelor specifice; evaluarea unor instrumente sau realizări, prelucrarea și interpretarea unor rezultate	• Chestionar scris • Răspuns oral • Caiet de laborator, lucrări experimentale, referate, examinare de laborator, etc. • Demonstrație practică			CPE
11.4d Proiect	• Calitatea proiectului realizat, corectitudinea documentației proiectului, justificarea soluțiilor alese	• Autoevaluarea, prezentarea și/sau susținerea proiectului • Evaluarea critică a unui proiect		20% (minim 5)	nCPE
11.5 Standard minim de performanță ²⁸ Pentru promovarea examenului, candidatul trebuie să cunoască principiile OOP si modul de abstractizare si prelucrare a datelor. De asemenea, el trebuie să aibă capacitatea să utilizeze aceste principii necesare pentru rezolvarea diferitelor categorii de probleme. Pentru intrarea în examen, sunt necesare minim 10 notari la activitățile de laborator.					

Fișa disciplinei cuprinde componente adaptate persoanelor cu CES (persoane cu dizabilități și persoane cu potențial înalt), în funcție de tipul și gradul acestora, la nivelul tuturor elementelor curriculare (competențe, obiective, conținuturi, metode de predare, evaluare alternativă), pentru a asigura șanse echitabile în pregătirea academică a tuturor studenților, acordând atenție sporită nevoilor individuale de învățare.

Data completării: | 1 | 9 | / | 0 | 9 | / | 2 | 0 | 2 | 5 |

Data avizării în Departament: | 2 | 3 | / | 0 | 9 | / | 2 | 0 | 2 | 5 |

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Lector Univ. Dr. Oana-Adriana ȚICLEANU	
Responsabil program de studii	Lect. univ. dr. Andreea SOLOMON	
Director Departament	Prof. Univ. Dr. Mugur Alexandru ACU	

¹ Licență / Master

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

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

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

⁵ Regim disciplină: O=Disciplină obligatorie; A=Disciplină opțională; U=Facultativă

⁶ Categoria formativă: S=Specialitate; F=Fundamentală; C=Complementară; I=Asistată integral; P=Asistată parțial; N=Neasistată

⁷ Este egal cu 14 săptămâni x numărul de ore de la punctul 3.1 (similar pentru 3.2.a.b.c.d.e.)

⁸ Liniile de mai jos se referă la studiul individual; totalul se completează la punctul 3.37.

⁹ Între 7 și 14 ore

¹⁰ Între 2 și 6 ore

¹¹ Suma valorilor de pe liniile anterioare, care se referă la studiul individual.

¹² Suma (3.5.) dintre numărul de ore de activitate didactică directă (NOAD) și numărul de ore de studiu individual (NOSI) trebuie să fie egală cu numărul de credite alocate disciplinei (punctul 3.7) x nr. ore pe credit (3.6.)

¹³ Numărul de credit se calculează după formula următoare și se rotunjește la valori vecine întregi (fie prin micșorare fie prin majorare)

$$\text{Nr. credite} = \frac{\text{NOCpSpD} \times C_C + \text{NOApSpD} \times C_A}{\text{TOCpSpD} \times C_C + \text{TOApSpD} \times C_A} \times 30 \text{ credite}$$

Unde:

- NOCpSpD = Număr ore curs/săptămână/disciplina pentru care se calculează creditele
- NOApSpD = Număr ore aplicații (sem./lab./pro.)/săptămână/disciplina pentru care se calculează creditele
- TOCpSpD = Număr total ore curs/săptămână din plan
- TOApSpD = Număr total ore aplicații (sem./lab./pro.)/săptămână din plan
- C_C/C_A = Coeficienți curs/aplicații calculate conform tabelului

Coeficienți	Curs	Aplicații (S/L/P)
Licență	2	1
Master	2,5	1,5
Licență lb. străină	2,5	1,25

¹⁴ Se menționează disciplinele obligatoriu a fi promovate anterior sau echivalente

¹⁵ Tablă, videoproiector, flipchart, materiale didactice specifice, platforme on-line etc.

¹⁶ Tehnică de calcul, pachete software, standuri experimentale, platforme on-line etc.

¹⁷ Rezultatele învățării se vor menționa în funcție de standardele specifice ale Comisiilor de specialitate ARACIS (<https://www.aracis.ro/ghiduri/>)

¹⁸ Titluri de capitole și paragrafe

¹⁹ Expunere, prelegere, prezentare la tablă a problematicei studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

²⁰ Discuții, dezbateri, prezentare și/sau analiză de lucrări, rezolvare de exerciții și probleme etc.

²¹ Demonstrație practică, exercițiu, experiment etc.

²² Studiu de caz, demonstrație, exercițiu, analiza erorilor etc.

²³ Alte tipuri de activități practice specifice

²⁴ Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

²⁵ CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală; N/A – nu se aplică

²⁶ Se va preciza numărul de teste și săptămânile în care vor fi susținute.

²⁷ Cercuri științifice, concursuri profesionale etc.

²⁸ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.