

FIȘA DISCIPLINEI

Anul universitar 2023 - 2024

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
1.2. Facultatea	Inginerie
1.3. Departament	Inginerie Industrială și management
1.4. Domeniul de studiu	Ingineria Transporturilor
1.5. Ciclul de studii ¹	Licență
1.6. Specializarea	Ingineria Transporturilor și a Traficului

2. Date despre disciplină

2.1. Denumirea disciplinei	Programarea calculatoarelor și Limbaje de Programare		Cod	FING.IIM.ITT.L.FO.2.1020.C-4.2	
2.2. Titular activități de curs	Prof. univ. dr. ing. Sorin Ioan Borza				
2.3. Titular activități practice	Conf.dr.ing. Inta Marinela				
2.4. An de studiu ²	1	2.5. Semestrul ³	2	2.6. Tipul de evaluare ⁴	colocviu
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
1		2			3
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
14		28			42
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					10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					14
Tutoriat ⁹					10
Examinări ¹⁰					4
3.3. Total ore alocate studiului individual¹¹ (NOSIsem)					58
3.4. Total ore din Planul de învățământ (NOADsem)					42
3.5. Total ore pe semestru¹² (NOADsem + NOSIsem)					100
3.6. Nr ore / ECTS					25
3.7. Număr de credite¹³					4

4. Precondiții (acolo unde este cazul)

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

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului ¹⁵	Tablă, videoproiector, laptop, elemente multimedia
5.2. De desfășurare a activităților practice (lab/sem/pr/aplic) ¹⁶	Sistem de calcul compus din rețea de calculatoare, videoproiector, ecran, tablă

6. Competențe specifice acumulate¹⁷

		Număr de credite alocat disciplinei ¹⁸	Repartizare credite pe competențe ¹⁹
6.1. Competențe profesionale	CP1		
	CP2		
	CP3		
	CP4		
	CP5		
	CP6		
6.2. Competențe transversale	CT1		
	CT2		
	CT3		

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

7.1. Obiectivul general	Disciplina PROGRAMAREA CALCULATOARELOR SI LIMBAJE DE PROGRAMARE II are ca scop învățarea și însușirea de către studenți a programării orientate obiect. În acest sens cursul cuprinde noțiuni de programare avansate, într-unul din cele mai populare limbaje de nivel înalt Visual Basic for Application.
7.2. Obiectivele specifice	Vor cunoaște și înțelege limbajul de programare VBA utilizarea lui in automatizarea aplicatiilor continute în diverse pachete software); Vor putea interpreta, automatiza si utiliza limbajul VBA pentru diverse aplicatii. Vor putea interpreta și explica funcționarea programelor scrise în limbajul VBA, in conjunctie ASP si cu tehnologia ADO pentru realizarea aplicatiilor distribuite; Vor deprinde cunoștințe solide privind programarea obiectuală în general.

8. Conținuturi

8.1. Curs ²⁰		Metode de predare ²¹	Nr. ore
Curs 1	Noțiuni fundamentale despre despre programarea orientată obiect. Obiect si clase de obiecte in C/C++	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Curs 2	Noțiuni fundamentale despre despre programarea orientată obiect. Obiect si clase de obiecte in C/C++	explicația; prelegerea intensificată, folosind metode multimedia;	2



		acțiuni interactive; organizator grafic.	
Curs 3	Tipuri abstracte de date implementate folosind clase de obiecte Instanțiere DynamicArray pentru diferite tipuri de date Diagrame UML	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Curs 4	Tipuri abstracte de date implementate folosind clase de obiecte Instanțiere DynamicArray pentru diferite tipuri de date Diagrame UML	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Curs 5	Metode ale claselor si obiectelor Moștenire Suprascrierea Polimorfism Legarea dinamica	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Curs 6	Operatii de intrare /ieșire Ierarhii de clase Fisiere header din IO Stream Input / Output File Stream Tratarea exeptiilor	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Curs 7	Containere cu elemente generice Creare si manipulare Adaptor de container Container asociativ Șabloane	explicația; prelegerea intensificată, folosind metode multimedia; acțiuni interactive; organizator grafic.	2
Total ore curs:			14



Activități practice (8.2.a. Seminar ²² / 8.2.b. Laborator ²³ / 8.2.c. Proiect ²⁴)	Metode de predare	Nr. ore
Act.1 Clase si obiecte în C/C++ Declarații de obiecte Metode Protecția atributelor și a metodelor Constructori Obiecte ca parametri de funcție	Conversația Studii de caz Rezolvare probleme individuale	2
Act.2 Clase si obiecte VBA Declarații de obiecte Metode Protecția atributelor și a metodelor Constructori Obiecte ca parametri de funcție	Conversația Studii de caz Rezolvare probleme individuale	2
Act.3 Clase si obiecte în VBA Declarații de obiecte Metode Protecția atributelor și a metodelor Constructori Obiecte ca parametri de funcție	Conversația Studii de caz Rezolvare probleme individuale	2
Act.4 Tipuri abstracte de date implementate folosind clase obiecte Listă – implementată secvențial pe vector dinamic (Dynamic Array) Regula celor trei (rule of tree) Clase/Metode prietene (friends) Implementare Iterator Instanțiere DynamicArray pentru diferite tipuri de date Diagrame UML	Conversația Studii de caz Rezolvare probleme individuale	2
Act.5 Tipuri abstracte de date implementate folosind clase obiecte Listă – implementată secvențial pe vector dinamic (Dynamic Array) Regula celor trei (rule of tree) Clase/Metode prietene (friends) Implementare Iterator Instanțiere DynamicArray pentru diferite tipuri de date Diagrame UML	Conversația Studii de caz Rezolvare probleme individuale	2
Act.6 Tipuri abstracte de date implementate folosind clase obiecte Listă – implementată secvențial pe vector dinamic (Dynamic Array) Regula celor trei (rule of tree) Clase/Metode prietene (friends) Implementare Iterator Instanțiere DynamicArray pentru diferite tipuri de date Diagrame UML	Conversația Studii de caz Rezolvare probleme individuale	2
ACT. 7 Proprietăți Obiectuale Moștenire simplă. Clase derivate. Suprascriere (overwrite) ≠ Supraîncărcare (overload) Polimorfism Legare dinamică (Dynamic binding) Mecanism C++ pentru polimorfism Moștenire multiplă Funcții virtuale	Conversația Studii de caz Rezolvare probleme individuale	2
ACT. 8 Proprietăți Obiectuale	Conversația	2



Act. 8 Moștenire simplă. Clase derivate. Suprascriere (overwrite) ≠ Supraîncărcare (overload) Polimorfism Legare dinamică (Dynamic binding) Mecanism C++ pentru polimorfism Moștenire multiplă Funcții virtuale	Studii de caz Rezolvare probleme individuale	
ACT. 9 Operații I/O I/O streams. I/O Hierarchies of classes. Hierarhie de clase din biblioteca standard IO C++ Streamuri standard – definite în iostream Supraîncărcare operatori <> pentru tipuri utilizator Flag-uri Fișiere Tratarea excepțiilor în C++	Conversația Studii de caz Rezolvare probleme individuale	2
ACT. 10 Operații I/O I/O streams. I/O Hierarchies of classes. Hierarhie de clase din biblioteca standard IO C++ Streamuri standard – definite în iostream Supraîncărcare operatori <> pentru tipuri utilizator Flag-uri Fișiere Tratarea excepțiilor în C++	Conversația Studii de caz Rezolvare probleme individuale	2
Act.11 Qt Toolkit QT - Module și utilitare Instalare QT, eclipse plugin Qapplication Componente grafice QT (widgets) Layout management QtDesigner din Eclipse Creare de interfețe grafice vizual (folosind drag & drop)	Conversația Studii de caz Rezolvare probleme individuale	2
Act.12 Qt Toolkit QT - Module și utilitare Instalare QT, eclipse plugin Qapplication Componente grafice QT (widgets) Layout management QtDesigner din Eclipse Creare de interfețe grafice vizual (folosind drag & drop)	Conversația Studii de caz Rezolvare probleme individuale	2
Act.13 Containere cu elemente generice Șabloane de proiectare Standard Template Library (STL) Adaptor de containere (Container adaptors) Șablonul arhitectură stratificată - Gestiunea memoriei	Conversația Studii de caz Rezolvare probleme individuale	2
Act.14 Containere cu elemente generice Șabloane de proiectare Standard Template Library (STL) Adaptor de containere (Container adaptors) Șablonul arhitectură stratificată - Gestiunea memoriei	Conversația Studii de caz Rezolvare probleme individuale	2
Total ore seminar/laborator		28

8.2. Activități practice

9. Bibliografie

9.1. Referințe bibliografice recomandate	<ul style="list-style-type: none"> ▪ Mircea Dorin Popovici, Mircea Ioan Popovici – C++. Tehnologia orientata spre obiecte, ▪ Editura Teora, Bucuresti, 2002. ▪ Borza Sorin, Realizarea Aplicațiilor Microsoft Office folosind limbajul Visual Basic for Application, Ed. Universității « Lucian Blaga » Sibiu, 2011; ▪ Borza, S., Ință, M., Programarea obiectuală a bazelor de date Visual Fox, Ed. Universității din Sibiu, 2005. ▪ Bjarne Stroustrup - The C++ Language (second edition); Editura Addison Wesley, 1993; ▪ Margaretz Ellis, Bjarne Stroustrup - The Annotated C++ Reference Manual; Editura Addison Wesley, 1991;
9.2. Referințe bibliografice suplimentare	<p>Borland C++.Version 4.5 - Programmers Guide, Borland International, 1995;</p> <p> </p> <p> </p> <p> </p> <p> </p> <p> </p>

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului²⁵

- elaborarea unor strategii de îmbunătățire a funcțiilor cognitive din input, elaborare și output.
- elaborarea unor instrumente eficiente de cunoaștere a profesionalismului personal
- proiectarea și implementarea unor activități, proiecte de cercetare cu scopul aplicării competențelor dobândite în urma studiului disciplinei
- Se realizează prin discuții periodice în cadru formal și informal cu reprezentanții firmelor de profil

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare		11.3 Pondere din nota finală	Obs. ²⁶
11.4a Examen / Colocviu	<ul style="list-style-type: none"> • Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea) 	Teste pe parcurs ²⁷ :	30 %	70 % (minim 5)	
Teme de casă:	15 %	Alte activități ²⁸ :	5%		
Evaluare finală:	50 % (min. 5)				
11.4b Seminar	<ul style="list-style-type: none"> • Frecvența/relevanța intervențiilor sau răspunsurilor 	Evidența intervențiilor, portofoliu de lucrări (referate, sinteze științifice)			
11.4c Laborator	<ul style="list-style-type: none"> • Cunoașterea aparaturii, a modului de utilizare a instrumentelor specifice; evaluarea unor instrumente sau realizări, prelucrarea și interpretarea unor rezultate 	<ul style="list-style-type: none"> • Chestionar scris • Răspuns oral • Caiet de laborator, lucrări experimentale, referate etc. • Demonstrație practică 		30 % (minim 5)	
11.4d Proiect	<ul style="list-style-type: none"> • Calitatea proiectului realizat, corectitudinea 	<ul style="list-style-type: none"> • Autoevaluarea, prezentarea și/sau susținerea proiectului • Evaluarea critică a unui proiect 		% (minim 5)	



	documentației proiectului, justificarea soluțiilor alese			
11.5 Standard minim de performanță ²⁹ 50% rezultat după însumarea punctajelor ponderate conform pct.11.3.				

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_|_8_| / |_0_|_9_| / |_2_|_0_|_2_|_3_|

Data avizării în Departament: |_2_|_5_| / |_0_|_9_| / |_2_|_0_|_2_|_3_|

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Prof. univ. dr. ing. Sorin Ioan Borza	
Responsabil program de studii	Conf. Univ. dr. ing. Lucian Lobonț	
Director Departament	Prof. univ. dr. ing. Dan Dumitrașcu	



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.)

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)

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

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
- TOCpSdP = Număr total ore curs/săptămână din plan
- TOApSdP = 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 Competențele din Grilele aferente descrierii programului de studii, adaptate la specificul disciplinei

¹⁸ Din planul de învățământ

¹⁹ Creditele alocate disciplinei se distribuie pe competențe profesionale și transversale în funcție de specificul disciplinei

²⁰ Titluri de capitole și paragrafe

²¹ Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții (pentru fiecare capitol, dacă este cazul)

²² Discuții, dezbateri, prezentare și/sau analiză de lucrări, rezolvare de exerciții și probleme

²³ Demonstrație practică, exercițiu, experiment

²⁴ Studiu de caz, demonstrație, exercițiu, analiza erorilor etc.

²⁵ Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

²⁶ CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală; N/A – nu se aplică

²⁷ Se va preciza numărul de teste și săptămânile în care vor fi susținute.

²⁸ Cercuri științifice, concursuri profesionale etc.

²⁹ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.