

Anexa 2.

## FIȘA DISCIPLINEI\*

### 1. Date despre program

Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
Facultatea	Facultatea de Științe
Departament	Matematică – Informatică
Domeniul de studiu	Informatică
Ciclul de studii	Licență
Specializarea	Informatică

### 2. Date despre disciplină

Denumirea disciplinei	Programare orientată obiect			
Codul cursului	Tipul cursului	An de studiu	Semestrul	Număr de credite
380602102008	DF	1	2	5
Tipul de evaluare	Categororia formativă a disciplinei (DF=fundamentală.; DD=domeniu; DS=specialitate; DC=complementară)			
Examen	E			
Titular activități curs	Lector univ. dr. Mircea-Adrian Mușan			
Titular activități seminar / laborator/ proiect	Lector univ. dr. Costel Maniu			

### 3. Timpul total estimat

Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână				
Curs	Seminar	Laborator	Proiect	Total
2	-	2	-	4
Extinderea disciplinei în planul de învățământ – Total ore din planul de învățământ				
Curs	Seminar	Laborator	Proiect	Total (NOAD <sub>sem</sub> )
28	-	28	-	56

Distribuția fondului de timp pentru studiu individual		Nr.ore
Studiul după manual, suport de curs, bibliografie și notițe		24
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren		5
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri		23
Tutoriat:		14
Examinări:		3
Total ore alocate studiului individual (NOSI <sub>sem</sub> )		69
Total ore pe semestru (NOAD <sub>sem</sub> + NOSI <sub>sem</sub> )		125

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

De curriculum	Parcurserea disciplinei <i>Fundamentele programării</i>
De competențe	-



## 5. Condiții (acolo unde este cazul)

De desfășurare a cursului	-
De desfășurare a sem/lab/pr	-

## 6. Competențe specifice acumulate

Competențe profesionale	Cunoașterea și utilizarea adecvată a noțiunilor specifice tehnicii OOP. Capacitatea de a interpreta și înțelege programe și aplicații realizate, pe baza codului sursă. Capacitatea de a proiecta și realiza programe folosind tehnica OOP, pe baza specificațiilor.
Competențe transversale	Dezvoltarea atitudinii pozitive față de muncă și responsabilitate pentru propria pregătire profesională. Dezvoltarea spiritului de muncă în echipă.

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

Obiectivul general al disciplinei	Însușirea cunoștințelor generale despre paradigma programării orientate pe obiecte.
Obiectivele specifice	Însușirea unui limbaj care suportă paradigma de programare, ca instrument efectiv de lucru (limbajul C++). Elemente primare și fundamentale de proiectare. Introducerea în elementele programării vizuale prin intermediul limbajului Visual C++.

## 8. Conținuturi

Curs		Nr. ore
Curs 1	<b>Metoda programării orientată pe obiecte.</b> Contextul apariției metodei. Principii ale metodei de programare. Calitatea produselor program și noua metodă. Limbajele de programare care suportă metoda de programare orientată pe obiecte. Limbajul C++, superset al limbajului C. Tipul referință. Funcții in-line. Pointeri. Structs. Pointeri Smart.	2
Curs 2-3	<b>Introducere în programarea obiectuală. Clase și obiecte.</b> Tipul abstract de date. Încapsulare. Declararea și definirea unei clase. Declararea obiectelor și referirea membrilor. Structura spațiului obiectelor. Constructori generali. Constructori de inițializare prin copiere. Destructorii.	4
Curs 4	<b>Facilități avansate ale programării OPP în limbajul C++.</b> Membri statici. Funcții membru constante. Obiecte constante. Array-uri de obiecte. Supraîncărcarea operatorilor. Funcții și clase prietene.	2
Curs 5	<b>Realizarea operațiilor de intrare și ieșire prin intermediul obiectelor.</b> Clase și obiecte de intrare și ieșire. operații standard cu format. Controlul formatului în operații. Manipulatori. Operații fără format. Supraîncărcarea operatorilor de extracție și inserție pentru obiectele unei clase.	2
Curs 6	<b>Supraîncărcarea operatorilor remarcabili în C++.</b> Supraîncărcarea operatorului de atribuire. Supraîncărcarea operatorilor de relație. Supraîncărcarea operatorului de apel de funcție. Supraîncărcarea operatorului de	2



# ULBS

Universitatea "Lucian Blaga" din Sibiu

Ministerul Educației și Cercetării

Universitatea "Lucian Blaga" din Sibiu

Prorector Programe Academice

	indexare. Conversii. Supraîncărcarea operatorului de <i>cast</i> .	
Curs 7-9	<b>Principiul moștenirii în OOP.</b> Moștenirea simplă: controlul membrilor. Moștenirea multiplă: moștenirea repetată. Ierarhii de clase: constructorii și destructorii, supraîncărcarea funcțiilor, operatorul de atribuire, conversia obiectelor și a pointerilor la obiecte, redefinirea funcțiilor. Polimorfism și funcții virtuale în ierarhii de clase. Interapelarea funcțiilor în ierarhii de clase. Virtualizarea destructorilor în ierarhii de clase. Clase abstracte.	6
Curs 10	<b>Obiecte fișier în programarea orientată spre obiecte.</b> Modul de lucru cu obiecte fișier. Tratarea erorilor în lucrul cu fișier. Prelucrarea fișierelor de tip text. Realizarea lucrului cu fișiere binare: accesul secvențial și accesul direct.	2
Curs 11	<b>Genericitatea în programarea orientată obiect.</b> Funcții generice. Clase generice. Biblioteca STL de clase generice: principalele clase container. Algoritmi generici, funcții generice.	2
Curs 12	<b>Tratarea excepțiilor și portabilitatea tipurilor.</b> Tratarea excepțiilor. Instrucțiunea <i>throw</i> . Lansarea excepțiilor cu o funcție din cadrul blocului <i>try</i> . Aplicații ale tratării excepțiilor. Restricționarea și relansarea excepțiilor. Spații de nume.	2
Curs 13	<b>Elemente primare și fundamentale de proiectare.</b> Modelarea obiectelor: modelul obiectelor, modelul dinamic al obiectelor, modelul funcțional al obiectelor. Etapele activității de realizare a proiectului: analiza, proiectarea de ansamblu, proiectarea de detaliu.	2
Curs 14	<b>Inițiere în programarea concurentă.</b> Procese. Fire de execuție.	2
<b>Total ore curs:</b>		<b>28</b>
<b>Seminar/Laborator</b>		Nr. ore
Sem 1	<b>Utilizarea îmbunătățirilor limbajului C++ în programarea procedurală.</b> Realizarea unor programe care utilizează referințe în transmiterea parametrilor. Realizarea de programe de lucru cu memoria dinamică utilizând operatorii <i>new</i> și <i>delete</i> . Programe care utilizează funcții cu argumente implicite și realizează supraîncărcarea funcțiilor. Realizarea programelor ce utilizează funcții <i>in-line</i> și operatorul de rezoluție.	2
Sem 2-3	<b>Introducere în conceptul de OOP.</b> Aplicații ce construiesc clase, declară și alocă memorie pentru obiecte. Realizarea unor aplicații pentru exemplificarea instanțierii prin utilizarea constructorilor. Destructorii.	4
Sem 4	<b>Supraîncărcarea operatorilor aritmetici.</b> Realizarea aplicațiilor cu dezvoltare de clase ce implică funcții prieten și operatori aritmetici supraîncărcați.	2
Sem 5	<b>Realizarea de aplicații prin operații de intrare/ieșire în C++.</b> Clase de bază pentru streamuri. Utilizarea membrilor clasei <i>IOS</i> pentru formatarea operațiilor de I/O. Manipulatori. Supraîncărcarea operatorilor " <i>&lt;&lt;</i> " și " <i>&gt;&gt;</i> ".	2
Sem 6	<b>Realizarea de aplicații prin operatori speciali supraîncărcați în C++.</b> Pointerul <i>this</i> . Supraîncărcarea operatorului " <i>[]</i> ". Supraîncărcarea operatorului " <i>()</i> ". Supraîncărcarea operatorului " <i>,</i> ". Supraîncărcarea operatorului " <i>=</i> ". Supraîncărcarea operatorilor " <i>new</i> " și " <i>delete</i> ".	2



Sem 7	<b>Construcția de aplicații folosind conceptul de moștenirea simplă în tehnica OOP.</b> Controlul accesului la membrii moșteniți. Constructorii și destructorii în ierarhii de clase. Alte probleme generate de procesul moștenirii. Aplicații complexe utilizând procesul de moștenire simplă.	2
Sem 8	<b>Construcția de aplicații folosind moștenirea multiplă în OOP.</b> Clase virtuale. Funcții virtuale. Polimorfism.	2
Sem 9	<b>Clase abstracte și clase prieten.</b> Realizarea de aplicații complexe ce folosesc concepte precum: polimorfism, clase abstracte și clase prieten.	2
Sem 10	<b>Prelucrarea fișierelor text și binare în OOP.</b> Ierarhia de clase pentru lucrul cu fișiere. Deschiderea fișierelor. Închiderea fișierelor. Intrări și ieșiri la nivel de fișier. Accesul la datele dintr-un fișier. Realizarea de aplicații prin folosirea obiectelor fișier în tehnica OOP.	2
Sem 11	<b>Funcții generice. Clase generice.</b> Șabloane. Clase generice. Realizarea de exemple cu două clase generice.	2
Sem 12	<b>Realizarea de aplicații pentru tratarea excepțiilor.</b> Tratarea excepțiilor. Instrucțiunea <i>throw</i> . Lansarea excepțiilor cu o funcție din cadrul blocului <i>try</i> . Aplicații ale tratării excepțiilor. Spații de nume.	2
Sem 13	<b>Introducere în proiectare.</b> Studiu de caz.	2
Sem 14	<b>Prezentarea proiectului final realizat individual de către studenți.</b>	2
<b>Total ore seminar/laborator</b>		<b>28</b>

### Metode de predare

Explicația, exemplificarea, învățarea prin descoperire, studiu de caz, analiza și proiectarea problemei.

### Bibliografie

Referințe bibliografice recomandate	Valer Roșca, <i>Programarea orientată pe obiecte în C++</i> , Editura Universității "Lucian Blaga", Sibiu, 2005
	Mircea Mușan, <i>Programare orientată pe obiecte</i> , note de curs și laborator – în format electronic
Referințe bibliografice suplimentare	Liviu Negrescu, <i>Limbajele C și C++ pentru începători. Limbajul C++</i> , Editura Alabastră, Cluj-Napoca, 2002
	Herbert Schildt, <i>C++. Manual complet</i> , Editura Teora, București, 2000
	Dr. Kris Jamsa, Lars Klander, <i>Totul despre C și C++. Manual fundamental de programare în C și C++</i> , Editura Teora, București, 2000

### 9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului

Prin discuții și întâlniri periodice cu reprezentanții companiilor producătoare de software și componente pentru automotive, ce își bazează activitatea pe programarea obiectuală și aplicațiile acesteia.

### 10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Ponderea în nota finală	Obs.**
Curs	Test de evaluare finală	Testare scrisă	60%	CEF



# ULBS

Universitatea "Lucian Blaga" din Sibiu

Ministerul Educației și Cercetării

Universitatea "Lucian Blaga" din Sibiu

Prorector Programe Academice

Laborator	Testare de laborator	Teme de casă, Activitate, Test de laborator	40%	CEF
Standard minim de performanță				
Pentru promovarea examenului, trebuie obținută minim nota 5 la evaluările pe parcurs și la examenul de evaluare finală. Ca și cerințe minime de absolvire a examenului sunt: însușirea conceptelor de clasă și încapsulare, constructori și destructori și conceptul de moștenire simplă.				

(\*) Fișa disciplinei cuprinde componente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

(\*\*) CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală;

Data completării: xxx 09.2020

Data avizării în Departament: 25.09.2020

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Lector univ. dr. Mușan Mircea-Adrian	
Director de departament	Profesor univ. dr. Acu Mugur	