

FIȘA DISCIPLINEI

1. Date despre program

Instituția de învățământ superior	Universitatea <i>Lucian Blaga</i> din Sibiu
Facultatea	Facultatea de Inginerie
Departament	Departamentul de <i>Calculatoare și Inginerie Electrică</i>
Domeniul de studiu	Inginerie Electronică, Telecomunicații și Tehnologii Informaționale
Ciclul de studii	Studii de licență
Specializarea	Electronică Aplicată

2. Date despre disciplină

Denumirea disciplinei	Programare obiect-orientată			
Codul cursului	Tipul cursului	An de studiu	Semestrul	Număr de credite
EA.306.DO	Obligativu	2	1	5
Tipul de evaluare	Categoriza formativă a disciplinei (DF=fundamentală.; DD=domeniu; DS=specialitate; DC=complementară)			
Colocviu	DD			
Titular activități curs	Conf. dr. ing. Macarie BREAZU			
Titular activități seminar / laborator / proiect	Șef lucr. dr. ing. Ioana COFARU, Dr. ing. Mirela IOANEȘIU			

3. Timpul total estimat

Extinderea disciplinei în planul de învățământ – număr de ore pe săptămână				
Curs	Seminar	Laborator	Proiect	Total
2	0	2	1	5
Extinderea disciplinei în planul de învățământ – Total ore din planul de învățământ				
Curs	Seminar	Laborator	Proiect	Total (<i>NOAD_{sem}</i>)
28	0	28	14	70

Distribuția fondului de timp pentru studiu individual		Nr.ore
Studiul după manual, suport de curs, bibliografie și notițe		25
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren		10
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri		15
Tutoriat:		2
Examinări:		3
Total ore alocate studiului individual (<i>NOSI_{sem}</i>)		55
Total ore pe semestru (<i>NOAD_{sem} + NOSI_{sem}</i>)		125

4. Precondiții (acolo unde este cazul)

De curriculum	
De competențe	

5. Condiții (acolo unde este cazul)

De desfășurare a cursului	videoproiector
De desfășurare a sem/lab/pr	rețea de calculatoare

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> • Aplicarea cunoștințelor, conceptelor și metodelor de bază privitoare la arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare • Proiectarea și utilizarea unor aplicații hardware și software de complexitate redusă specifice electronicii aplicate • Aplicarea cunoștințelor, conceptelor și metodelor de bază din: electronică de putere, sisteme automate, gestionarea energiei electrice, compatibilitate electromagnetică
Competențe transversale	<ul style="list-style-type: none"> • Analiza metodică a problemelor întâlnite în activitate, identificând elementele pentru care există soluții consacrate, asigurând astfel îndeplinirea sarcinilor profesionale • Adaptarea la noile tehnologii, dezvoltarea profesională și personală, prin formare continuă folosind surse de documentare tipărite, software specializat și resurse electronice în limba română și, cel puțin, într-o limbă de circulație internațională

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

Obiectivul general al disciplinei	Înșușirea tehnicilor de programare orientată pe obiecte, utilizarea unor biblioteci de obiecte și a unor medii vizuale de dezvoltare a aplicațiilor având la bază programarea orientată pe obiecte.
Obiectivele specifice	<ul style="list-style-type: none"> • Cunoașterea și operarea adecvată cu noțiunile specifice disciplinei. • Dobândirea unei atitudini pozitive față de (necesitatea validării aspectelor teoretice prin) aplicația practică.

8. Conținuturi

Curs		Nr. ore
Curs 1	Introducere. Necesitatea programării orientate pe obiecte. Avantaje.	2
Curs 2	Elemente de programare neobiectuală: operatori, funcții, durată de viață și domeniu de vizibilitate, încapsulare la nivel de modul.	2
Curs 3	Elemente de programare neobiectuală: pointeri.	2
Curs 4	Încapsularea la nivel de clasă. Drepturi de acces. Constructori și destructori. Membrii statici.	2
Curs 5	Moștenirea. Drepturi de acces în clasa derivată. Constructori și destructori. Moștenirea multiplă.	2
Curs 6	Legare statică și dinamică. Implementarea polimorfismul folosind VMT. Clase abstracte. Tratarea uniformă a masivelor eterogene	2
Curs 7	Legare statică și dinamică. Implementarea polimorfismul folosind VMT. Clase abstracte. Tratarea uniformă a masivelor eterogene	2
Curs 8	Redefinirea operatorilor. Particularități.	2
Curs 9	Redefinirea operatorilor. Particularități.	2

Curs 10	Tratarea structurată a excepțiilor. Discriminarea excepțiilor. Tratarea polimorfică a excepțiilor. Achiziție și eliberare de resurse.	2
Curs 11	Studiu de caz: biblioteca de streamuri.	2
Curs 12	Studiu de caz: biblioteca de streamuri.	2
Curs 13	Functii și clase template.	2
Curs 14	Elemente de ingineria programării orientate pe obiecte. Limbajul UML	2
Total ore curs:		28
Laborator		Nr. ore
Lab 1	Noțiunea de structură. Apelul funcțiilor prin valoare, pointeri și referință	2
Lab 2	Încapsularea datelor și a metodelor în structură. Noțiunea de clasă.	2
Lab 3	Noțiunea de constructor / destructor. Metode cu parametrii impliciți.	2
Lab 4	Moștenirea, drepturi de acces în clasa derivată.	2
Lab 5	Polimorfism, clase abstracte.	2
Lab 6	Redefinire de operatori.	2
Lab 7	Streamuri, biblioteca de streamuri.	2
Lab 8	Particularitățile programării sub Windows. Modelul unei aplicații Windows.	2
Lab 9	Noțiunea de resursa și utilizarea acestora, utilizarea editorului de resurse.	2
Lab 10	Utilizarea mediului C++ Builder. Forme, obiecte, componente și evenimente.	2
Lab 11	Aplicații cu mai multe forme. Forme modale și nemodale.	2
Lab 12	Utilizarea componentelor pentru baze de date.	2
Lab 13	Aplicații cu baze de date. Gestionarea stării unei baze de date.	2
Lab 14	Evaluarea însușirii cunoștințelor pe baza unei aplicații complexe.	2
Total ore laborator:		28
Proiect		
Pr 1	Primirea temei. Analiza și documentarea cerințelor.	2
Pr 2	Planificarea proiectului. Realizarea documentelor „use-case”.	2
Pr 3	Implementare.	2
Pr 4	Implementare	2
Pr 5	Testare și depanare.	2
Pr 6	Realizarea documentației.	2
Pr 7	Predarea proiectului, evaluare.	2
Total ore proiect:		14

Metode de predare

Prelegeri, explicații, conversații, problematizări, demonstrații, studii de caz, exerciții.	Limba de predare	Română
---	------------------	--------

Bibliografie

Referințe bibliografice recomandate	1. Macarie Breazu – ”Programare orientată pe obiecte. Principii”, Editura Universității "Lucian Blaga" din Sibiu, Sibiu, 2002, cota biblioteca ULBS: 04/B82
	2. Macarie Breazu, Daniel Morariu, Gabriel Curea - ”Programare Orientată pe Obiecte - Îndrumar de laborator”, Editura Universității “Lucian Blaga” din Sibiu, ISBN 978-973-739-918-2, Sibiu, 2010, cota biblioteca ULBS: 04/B82

	3. Bertrand Meyer - " <i>Touch of Class : Learning to Program Well with Objects and Contracts</i> ", Springer, 2009, cota biblioteca ULBS: 04/M63
	4. *** - Documentația mediilor de programare folosite
Referințe bibliografice suplimentare	1. Bjarne Stroustrup – " <i>The C++ Programming Language. Second Edition</i> ", Addison Wesley Publishing Company, 1993, traducere Editura Teora, 2003, cota biblioteca ULBS: 04/S91
	2. Dan Roman – " <i>Ingineria programării obiectuale</i> ", Editura Albastră, Cluj-Napoca, 1996
	3. Ioan Jurcă – " <i>Programarea orientată pe obiecte în limbajul C++</i> ", Editura Eurobit, Timișoara, 1992, cota biblioteca ULBS: 04/J92

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

--

10. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Ponderea în nota finală	Obs.*
Curs	Evaluarea aspectelor teoretice	lucrare scrisă	30%	CEF
	Evaluarea aspectelor practice	lucrare scrisă	30%	CEF
Laborator	Realizare teme laborator	examinare orală	20%	CPE
Proiect	Realizare proiect	susținere orală	20%	CPE

Standard minim de performanță

Nota 4.50 pe fiecare componentă evaluată

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

Data completării: 4.09.2020

Data avizării în Departament:.....

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Conf. dr. ing. Macarie BREAZU	
Director de departament	Prof. dr. ing. Daniel VOLOVICI	