FIŞA DISCIPLINEI

Anul universitar 2022 - 2023

1. Date despre program

1.1. Instituția de învățământ superior	**Universitatea Lucian Blaga din Sibiu**
1.2. Facultatea	Facultatea de Inginerie
1.3. Departament	Departamentul de Calculatoare şi Inginerie Electrică
1.4. Domeniul de studiu	Inginerie Electrică
1.5. Ciclul de studii[1]	Licenţă
1.6. Specializarea	ELECTROMECANICĂ

2. Date despre disciplină

2.1. Denumirea disciplinei	Programarea calculatoarelor și limbaje de programare II	Cod	FING.CIE.EM.L.FO.3.2021.C-6.4
2.2. Titular activități de curs	conf. dr. ing. Macarie BREAZU		
2.3. Titular activități practice	dr. ing. Mirela IOANEȘIU		

2.4. An de studiu[2]	2	**2.5.** Semestrul[3]	3	**2.6.** Tipul de evaluare[4]	C
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	0	2	1	0	**5**
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	0	28	14	0	**70**

Distribuția fondului de timp pentru studiu individual[8]	**Nr. ore**
Studiul după manual, suport de curs, bibliografie și notițe	21
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren	21
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri	21
Tutoriat[9]	14
Examinări[10]	3
3.3. Total ore alocate studiului individual[11] (NOSIsem)	**80**
3.4. Total ore din Planul de învățământ (NOADsem)	**70**
3.5. Total ore pe semestru[12] (NOADsem + NOSIsem)	**150**
3.6. Nr ore / ECTS	**25**
3.7. Număr de credite[13]	**6**

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]	videoproiector
5.2. De desfășurare a activităților practice (lab/sem/pr/aplic)[16]	reţea de calculatoare

6. Competențe specifice acumulate[17]

		Număr de credite alocat disciplinei[18]	6
6.1. Competențe profesionale	CP1	Aplicarea adecvată a cunoştinţelor fundamentale de matematică, fizică, chimie specifice domeniului ingineriei electrice	Repartizare credite pe competențe[19]
	CP2	Operarea cu concepte fundamentale din ştiinţa calculatoarelor şi tehnologia informaţiei	4.00
	CP3	Aplicarea adecvată a cunoştinţelor privind conversia energetică, fenomenele electromagnetice şi mecanice specifice convertoarelor statice, electromecanice, echipamentelor electrice şi acţionărilor electromecanice	
	CP4	Utilizarea tehnicilor de măsurare a mărimilor electrice şi neelectrice şi a sistemelor de achiziţie de date în sistemele electromecanice	
	CP5	Automatizarea proceselor electromecanice	
	CP6	Realizarea activităţilor de exploatare, întreţinere, service, integrare de sistem	
6.2. Competențe transversale	CT1	Identificarea obiectivelor de realizat, a resurselor disponibile, condiţiilor de finalizare a acestora, etapelor de lucru, timpilor de lucru, termenelor de realizare aferente şi riscurilor aferente	0.50
	CT2	Identificarea rolurilor şi responsabilităţilor într-o echipă pluridisciplinară şi aplicarea de tehnici de relaţionare şi muncă eficientă în cadrul echipei	0.50
	CT3	Utilizarea eficientă a surselor informaţionale şi a resurselor de comunicare şi formare profesională asistată (portaluri Internet, aplicaţii software de specialitate, baze de date, cursuri on-line etc.) atât in limba romana cat si intr-o limba de circulaţie internationala	1.00

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

7.1. Obiectivul general	Însuşirea tehnicilor de programare orientată pe obiecte, utilizarea unor biblioteci de obiecte şi a unor medii vizuale de dezvoltare a aplicaţiilor având la bază programarea orientată pe obiecte.
7.2. Obiectivele specifice	• Cunoaşterea şi operarea adecvată cu noţiunile specifice disciplinei. • Dobândirea unei atitudini pozitive faţă de (necesitatea validării aspectelor teoretice prin) aplicaţia practică.

8. Conținuturi

8.1. Curs[20]		**Metode de predare**[21]	**Nr. ore**
Curs 1	Introducere. Necesitatea programării orientate pe obiecte. Avantaje.	Expunere	2
Curs 2	Elemente de programare neobiectuală: operatori, funcţii, durată de viata şi domeniu de vizibilitate, încapsulare la nivel de modul.	Expunere	2
Curs 3	Elemente de programare neobiectuală: pointeri.	Expunere	2
Curs 4	Încapsularea la nivel de clasă. Drepturi de acces. Constructori şi destructori. Membrii statici.	Expunere	2
Curs 5	Moştenirea. Drepturi de acces în clasa derivată. Constructori şi destructori. Moştenirea multiplă.	Expunere	2
Curs 6	Legare statică şi dinamică. Implementarea polimorfismul folosind VMT. Clase abstracte. Tratarea uniformă a masivelor eterogene	Expunere	2
Curs 7	Legare statică şi dinamică. Implementarea polimorfismul folosind VMT. Clase abstracte. Tratarea uniformă a masivelor eterogene	Expunere	2
Curs 8	Redefinirea operatorilor. Particularităţi.	Expunere	2
Curs 9	Redefinirea operatorilor. Particularităţi.	Expunere	2
Curs 10	Tratarea structurată a excepţiilor. Discriminarea excepţiilor. Tratarea polimorfică a excepţiilor. Achiziţie şi eliberare de resurse.	Expunere	2
Curs 11	Studiu de caz: biblioteca de streamuri.	Expunere	2
Curs 12	Studiu de caz: biblioteca de streamuri.	Expunere	2
Curs 13	Functii şi clase template.	Expunere	2
Curs 14	Elemente de ingineria programării orientate pe obiecte. Limbajul UML	Expunere	2
		Total ore curs:	**28**

8.2. Activități practice

8.2.b. Laborator		**Metode de predare**[22]	**Nr. ore**
Laborator 1	Noţiunea de structură. Apelul funcţiilor prin valoare, pointeri şi referinţă.	Exercițiu	2
Laborator 2	Încapsularea datelor şi a metodelor în structură. Noţiunea de clasă.	Exercițiu	2
Laborator 3	Noţiunea de constructor / destructor. Metode cu parametrii impliciţi.	Exercițiu	2
Laborator 4	Moştenirea, drepturi de acces în clasa derivată.	Exercițiu	2
Laborator 5	Polimorfism, clase abstracte.	Exercițiu	2
Laborator 6	Redefinire de operatori.	Exercițiu	2
Laborator 7	Streamuri, biblioteca de streamuri.	Exercițiu	2
Laborator 8	Particularităţile programării sub Windows. Modelul unei aplicaţii Windows.	Exercițiu	2
Laborator 9	Noţiunea de resursa şi utilizarea acestora, utilizarea editorului de resurse.	Exercițiu	2
Laborator 10	Utilizarea mediului C++ Builder. Forme, obiecte, componente şi evenimente.	Exercițiu	2
Laborator 11	Aplicaţii cu mai multe forme. Forme modale şi nemodale.	Exercițiu	2
Laborator 12	Utilizarea componentelor pentru baze de date.	Exercițiu	2
Laborator 13	Aplicaţii cu baze de date. Gestionarea stării unei baze de date.	Exercițiu	2
Laborator 14	Evaluarea însuşirii cunoştinţelor pe baza unei aplicaţii complexe.	Exercițiu	2
		Total ore laborator	28

8.2.c. Proiect		**Metode de predare[23]**	**Nr. ore**
Proiect 1	Primirea temei. Analiza si documentarea cerințelor.	Studiu de caz	2
Proiect 2	Planificarea proiectului. Realizarea documentelor „use-case".	Studiu de caz	2
Proiect 3	Implementare.	Studiu de caz	2
Proiect 4	Implementare	Studiu de caz	2
Proiect 5	Testare şi depanare.	Studiu de caz	2
Proiect 6	Realizarea documentaţiei.	Studiu de caz	2
Proiect 7	Predarea proiectului, evaluare.	Studiu de caz	2
		Total ore proiect	14

9. Bibliografie

9.1. Referințe bibliografice recomandate	1. Macarie Breazu – "*Programare orientată pe obiecte. Principii*", Editura Universităţii "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 Universităţii "Lucian Blaga" din Sibiu, ISBN 978-973-739-918-2, Sibiu, 2010, cota biblioteca ULBS: 04/B82
	3. Bertrand Meyer - "*Touch of Class : Learning to Program Well with Objects and Contracts*", Springer, 2009, cota biblioteca ULBS: 04/M63
	4. *** - Documentaţia mediilor de programare folosite
9.2. Referințe bibliografice suplimentare	1. Bjarne Stroustrup – "*The C++ Programming Language. Second Edition*", Addison Wesley Publishing Company, 1993, traducere Editura Teora, 2003, cota biblioteca ULBS: 04/S91
	2. Dan Roman – "*Ingineria programării obiectuale*", Editura Albastră, Cluj-Napoca, 1996
	3. Ioan Jurcă – "*Programarea orientată pe obiecte în limbajul C++*", Editura Eurobit, Timişoara, 1992, cota biblioteca ULBS: 04/J92

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului[24]

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.[25]
11.4a Examen / Colocviu	• Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea)	Teste pe parcurs[26]:	0%	60%	CEF
		Teme de casă:	0%		
		Alte activități[27]:	0%		
		Evaluare finală:	100%		
11.4c Laborator	• Cunoașterea aparaturii, a modului de utilizare a instrumentelor specifice; evaluarea unor	• Chestionar scris • Răspuns oral • Caiet de laborator, lucrări experimentale, referate etc.		20%	CPE

	instrumente sau realizări, prelucrarea și interpretarea unor rezultate	• Demonstrație practică		
11.4d Proiect	• Calitatea proiectului realizat, corectitudinea documentației proiectului, justificarea soluțiilor alese	• Autoevaluarea, prezentarea și/sau susținerea proiectului • Evaluarea critică a unui proiect	20%	CPE
11.5 Standard minim de performanță[28] Nota 4.50 pe fiecare componentă evaluată				

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: 07.09.2022

Data avizării în Departament: 09.09.2022

	Grad didactic, titlul, prenume, numele	**Semnătura**
Titular disciplină	conf. dr. ing. Macarie BREAZU	
Responsabil program de studii	conf. dr. ing. Mihai BOGDAN	
Director Departament	prof. dr. ing. Daniel VOLOVICI	
Decan	prof. dr. ing. Sever-Gabriel RACZ	

[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 alocat 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*

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

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] *Demonstrație practică, exercițiu, experiment*
[23] *Studiu de caz, demonstrație, exercițiu, analiza erorilor etc.*
[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.*