

FIȘA DISCIPLINEI

Anul universitar 2023-2024

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 și Inginerie Electrică
1.4. Domeniul de studiu	Calculatoare și Tehnologia Informației
1.5. Ciclul de studii ¹	Licență
1.6. Specializarea	TEHNOLOGIA INFORMAȚIEI

2. Date despre disciplină

2.1. Denumirea disciplinei	Bazele programării calculatoarelor	Cod	TI.106.FO
2.2. Titular activități de curs	ș.l. dr. ing. Ileana Ioana COFARU		
2.3. Titular activități practice	ș.l. dr. ing. Ioana Cofaru		
2.4. An de studiu ²	1	2.5. Semestrul ³	1
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					28
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					9
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					14
Tutoriat ⁹					14
Examinări ¹⁰					4
3.3. Total ore alocate studiului individual¹¹ (NOSIsem)					69
3.4. Total ore din Planul de învățământ (NOADsem)					56
3.5. Total ore pe semestru¹² (NOADsem + NOSIsem)					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	Cunoștințe de bază de matematică. Nu sunt necesare alte cunoștințe prealabile. Nu sunt obligatorii cunoștințe de programare.
4.2. Competențe	

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului ¹⁵	Participare activă, lectura suportului de curs Tablă, videoproiector
5.2. De desfășurare a activităților practice (lab/sem/pr/aplic) ¹⁶	Elaborarea și susținerea lucrărilor planificate. Sală dotată cu calculatoare cu sistem de operare Windows.

6. Competențe specifice acumulate¹⁷

		Număr de credite alocate disciplinei ¹⁸	5	Repartizare credite pe competențe ¹⁹
6.1. Competențe profesionale	CP1	Operarea cu fundamente științifice, ingineresti și ale informaticii		2
	CP2	Proiectarea componentelor hardware, software și de comunicații		1
	CP3	Soluționarea problemelor folosind instrumentele științei și ingineriei calculatoarelor		1
	CP4	Proiectarea și integrarea sistemelor informatice utilizând tehnologii și medii de programare		
	CP5	Întreținerea și exploatarea sistemelor hardware, software și de comunicații		
	CP6	Utilizarea sistemelor inteligente		
6.2. Competențe transversale	CT1	Comportarea onorabilă, responsabilă, etică, în spiritul legii pentru a asigura reputația profesiei		
	CT2	Identificarea, descrierea și derularea proceselor din managementul proiectelor, cu preluarea diferitelor roluri în echipă și descrierea clară și concisă, verbal și în scris, în limba română și într-o limbă de circulație internațională, a rezultatelor din domeniul de activitate		0.5
	CT3	Demonstrarea spiritului de inițiativă și acțiune pentru actualizarea cunoștințelor profesionale, economice și de cultură organizațională		0.5

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

7.1. Obiectivul general	<ul style="list-style-type: none"> • Cunoașterea și înțelegerea principiilor generale ale disciplinei • Cunoașterea și operarea adecvată cu noțiunile specifice disciplinei • Identificarea principalelor surse de informare • Analiza critică a modelelor teoretice, ideilor și a abordărilor consacrate • Aptitudini de realizare a unei teme și a unui raport aferent • Dezvoltarea abilităților de cercetare individuală
7.2. Obiectivele specifice	<ul style="list-style-type: none"> • Înțelegerea principiilor de bază ale structurii sistemelor de calcul • Proprietățile algoritmilor, reprezentarea acestora prin scheme logice și prin pseudocod, respectarea principiilor programării structurate în procesul de elaborare a algoritmilor • Identificarea tipurilor de informații reprezentate într-un calculator; reprezentarea informațiilor; conversii de valori între sisteme de numerație; aritmetică binară; operații logice, legile lui De Morgan • Identificarea circuitelor digitale combinaționale și secvențiale care compun o microarhitectură generică (LC-3): tranzistoare CMOS, porți logice

	<p>(fundamentale), structuri logice combinaționale (Multiplexoare, Decodificatoare, Sumatoare), structuri logice secvențiale (bistabili, registre, memorii, automate).</p> <ul style="list-style-type: none"> • Înțelegerea modelului de execuție von Neumann, (componentele modelului și ciclul instrucțiunii) aferent unei arhitecturi simple LC-3, a limbajului de asamblare propriu. Cunoașterea interfeței cu tastatura și monitorul, a apelurilor sistem (întreruperi software reprezentând servicii ale sistemului de operare prin instrucțiuni dedicate), modurile de adresare pentru instrucțiunile cu referire la memorie (load / store), mecanismul de apel și revenire din subrutină. • Exemplificarea la nivelul limbajului C și implicațiile în hardware a structurilor de control, a diverselor tehnici de alocare a memoriei, lucrul cu stiva, apel de funcții și revenire, stivele de date aferente funcțiilor, transmiterea parametrilor în cazul apelului, recursivitate, pointeri, tablouri și structuri elementare de date.
--	--

8. Conținuturi

8.1. Curs ²⁰		Metode de predare ²¹	Nr. ore
Curs 1	Introducere în sistemele de procesare a informației. Calculatorul – dispozitiv universal de calcul. Structură. Nivele de transformare. Nevoia de abstractizare și "Când eșuează abstractizarea?".	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 2	Reprezentarea informațiilor. Entropia informațională. Exemplificare diferența dată – informație. Sisteme de numerație. Conversii de valori între sisteme de numerație. Coduri pentru detectarea și corectarea erorilor. Aritmetică binară. Operații logice. Depășirea domeniului de reprezentare. Extensia semnului.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 3	Algoritmi. Caracteristicile algoritmilor. Reprezentarea algoritmilor prin scheme logice și prin pseudocod.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 4	Structuri logice digitale. Tranzistori. Porți logice. Circuite logice combinaționale (Decodicatorul, Multiplexorul, Sumatorul). Proiectarea unei unități aritmetico-logice.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 5	Elemente primare de memorare. Bistabili. Regiștrii. Circuite numărătoare. Memorii. Configurarea unor memorii folosind circuite primare. Automate secvențiale și programabile. Automate cu număr finit de stări.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 6	Modelul arhitectural von Neumann. Componente de bază. Principiile procesării instrucțiunilor. Ciclul instrucțiunii Ciclul instrucțiunii văzut ca un automat cu număr finit de stări. Exemplificare pe simulatorul LC3 Microarchitecture Simulator („a graphical simulator for the LC3 processor, concentrating on the micro-architecture level”). Tipuri de instrucțiuni. Ceasul procesorului. Oprirea sistemului de calcul.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 7	LC-3 – Arhitectura setului de instrucțiuni. Calea fluxului de date. Organizarea memoriei la LC-3.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 8	Limbajul de asamblare aferent arhitecturii LC-3. Asamblorul. Etapele generării codului mașină. Tabela de simboluri.	Expunere, prelegere, prezentare la tablă a problematicii studiate,	2

		utilizare videoproiector, discuții cu studenții	
Curs 9	Apeluri de subrutine – directe și indirecte. Reveniri. Întreruperi software la nivel low. Salvarea și restaurarea regiștrilor. Strategiile “caller-save” respectiv “callee save”.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 10	Stiva – structură. Principiu de funcționare. Operații aferente (Push & Pop). Implementare hardware și software. Modul de lucru prin întreruperi hardware. Operații aritmetice folosind stiva. Rolul stivei în tratarea întreruperilor hardware imbricate.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 11	Generarea codului obiect pentru o arhitectură dată. Implementarea în hardware a funcțiilor din programele de nivel înalt. Stiva de date aferentă funcțiilor. Transferul parametrilor în cazul apelului. Translatarea codului de la nivel high (C) la nivel low (asamblare – LC3).	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 12	Introducere în recursivitate. Recursivitate vs. Recurență. Rolul stivei în implementarea recursivității. Comparația dintre recursiv și iterativ în alegerea algoritmului de rezolvare a problemelor. Avantaje / dezavantaje. Implementarea recursivității la nivelul stivei de date. Eroarea Stack Overflow. Tipuri de funcții recursive. Eliminarea recursivității.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 13	Pointeri. Tablouri. Transferul parametrilor prin adresă. Pointeri spre funcții. Legătura dintre nivelul high și low văzută prin intermediul modurilor de adresare. Alocarea și accesarea indirectă de variabile.	Expunere, prelegere, prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Curs 14	Recapitulare finală. Parcurgerea conceptelor esențiale.	Prezentare la tablă a problematicii studiate, utilizare videoproiector, discuții cu studenții	2
Total ore curs:			28

8.2. Activități practice

8.2.a. Seminar		Metode de predare ²²	Nr. ore
Seminar 1	Reprezentarea numerelor întregi: fără semn, cu semn (MS, C1, C2). Conversii din diverse baze. Reprezentarea numerelor fracționare: standardul IEEE 754. Operații aritmetico-logice în binar. Coduri ASCII și Unicode. Conversia din caracter în numeric și invers.	Rezolvare de exerciții și probleme.	4
Seminar 2	Algoritmi. Caracteristicile algoritmilor. Reprezentarea algoritmilor prin scheme logice și prin pseudocod.	Rezolvare de exerciții și probleme.	2
Seminar 3	Proiectarea circuitelor digitale folosind tranzistoare. Operații logice folosind porți logice fundamentale (NAND, NOR). Implementarea multiplexoarelor, sumatoarelor, decodificatoarelor.	Rezolvare de exerciții și probleme.	2
Seminar 4	Proiectarea unor circuite de memorie folosind circuite primare. Aplicații cu automate secvențiale și programabile. Utilizarea simulatorului memory.exe, care descrie operațiile de citire – scriere dintr-o memorie cu 4 locații având 3 biți de date fiecare.	Rezolvare de exerciții și probleme.	2
Seminar 5	Modelul de procesare „von Neumann”. Codificarea instrucțiunilor. Determinarea numărului de instrucțiuni executate pe un anumit microprocesor într-o secundă.	Rezolvare de exerciții și probleme.	2



	Determinarea numărului de accese la memorie pentru fiecare tip de instrucțiune. Utilizarea LC-3 microarhitectur.		
Seminar 6	Examinare parțială din materia parcursă.	Rezolvare de exerciții și probleme.	2
Seminar 7	Limbajul de asamblare aferent arhitecturii LC-3. Etapele generării codului mașină. Tabela de simboluri.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj de asamblare folosind simulatorul LC-3.	2
Seminar 8	Aplicații în LC-3 cu întreruperi software și apeluri imbricate de subrutine.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj de asamblare folosind simulatorul LC-3.	2
Seminar 9	Aplicații cu stiva și operațiile Push / Pop. Codificarea ASCII to Binary și Binary to ASCII în LC-3.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj de asamblare folosind simulatorul LC-3.	2
Seminar 10	Aplicații scrise recursiv și iterativ. Avantaje / dezavantaje. Implementarea recursivității la nivelul stivei de date. Evidențierea erorii Stack Overflow.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj C.	2
Seminar 11	Aplicații cu pointeri și tablouri. Transferul parametrilor prin adresa. Pointeri spre funcții. Legătura dintre nivelul high și low văzută prin intermediul modurilor de adresare. Alocarea și accesarea indirectă de variabile.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj C.	2
Seminar 12	Examinare parțială din materia parcursă.	Rezolvare de exerciții și probleme.	2
Seminar 13	Ghid de utilizare LC-3 Edit și LC-3 Simulate. Implementarea software a aplicațiilor anterior prezentate la curs și seminar. Depanarea și simularea folosind cele doua utilitare.	Folosirea calculatoarelor pentru rezolvarea unor probleme de programare în limbaj de asamblare folosind simulatorul LC-3.	2
Seminar 14	-	-	
Total ore seminar			28

9. Bibliografie

9.1. Referințe bibliografice recomandate	Patt Yale, Patel Sanjay – „Introduction to Computing Systems: from bits & gates to C & beyond”, McGraw-Hill Higher Education, 2003.
	Patterson David, Hennessy John – Computer Organisation and Design: The Hardware/Software Interface, Third Edition, Elsevier, 2005, ISBN: 1-55860-604.
	Cormen, T. H., Leiserson, C. E., Rivest, R. L. – Introduction to Algorithms. McGraw-Hill, New York, 1990.
	Pop Vasile – Analiza și sinteza dispozitivelor numerice. Curs litografiat, Institutul Politehnic Timișoara, 1986, vol. I și II.
	Vințan N. Lucian, Fundamente ale arhitecturii microprocesoarelor, Editura Matrix Rom, București, ISBN 978-606-25-0276-8, 2016 (547 pg.)
	Mârșanu Radu – „Sisteme de calcul”, Manual pentru licee de informatică, cls. IX-a, Editura Didactică și Pedagogică, București, 1996.
	Florea Adrian – Introducere în Știința și Ingineria Calculatoarelor. Interfața Hardware – Software, Editura Matrix ROM, București, ISBN 978-973-755-264-8, 2007 (313 pg.).
9.2. Referințe bibliografice suplimentare	Zaharia Mihai, Leon Florin – „Limbajul C de la Zero la Student”, Editura Politehnicum Iasi, 2004.
	Dana Lica, ș.a. – Fundamentele programării, Editura L&S Soft, București.
	Florea Adrian – Predicția dinamică a valorilor în microprocesoarele generației următoare, Editura MatrixROM, 2005.
	Golomety A. – Proiectarea Translatoarelor, Editura Universității "Lucian Blaga", Sibiu, 1997.
	Gorgan D., Sebestyen-Pal Gh., Computer Design, Editura albastra, Cluj-Napoca, ISBN 973-650-123-X, 2005.

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului²³

Disciplina își propune formarea unei gândiri critice, algoritmice, înțelegerii structurii sistemelor de calcul și a interacțiunii dintre componentele acestuia, punând bazele abilităților de proiectare necesare viitorilor ingineri. 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. ²⁴
11.4a Examen / Colocviu	<ul style="list-style-type: none"> Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea) 	Teste pe parcurs ²⁵ :	25%	60%	CPE
		Teme de casă:	10%		
		Alte activități ²⁶ :	5%		
		Evaluare finală:	60%		
11.4b Seminar	<ul style="list-style-type: none"> Frecvența/relevanța intervențiilor sau răspunsurilor 	Evidența intervențiilor, portofoliu de lucrări (referate, sinteze științifice)		40%	CPE
11.5 Standard minim de performanță ²⁷ <ul style="list-style-type: none"> Cunoașterea, înțelegerea și explicarea noțiunilor elementare specifice științei și ingineriei calculatoarelor. Interes constant manifestat pentru însușirea disciplinei. Îndeplinirea condițiilor minime obligatorii (50%) în privința temelor de casă, referatelor și a testelor date pe parcursul semestrului. 					CPE

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: 14.09.2023

Data avizării în Departament: 15.09.2023

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	ș.l. dr. ing. Ileana Ioana COFARU	
Responsabil program de studii	conf. dr. mat. Radu George CREȚULESCU	
Director Departament	prof. dr. ing. Daniel VOLOVICI	
Decan	prof. dr. ing. Maria VINȚAN	

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 credite se calculează după formula următoare și se rotunjește la valori vecine întregi (fie prin micșorare fie prin majorare)

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

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 obligatorii 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 Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

²⁴ CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală; N/A – nu se aplică

²⁵ Se va preciza numărul de teste și săptămânile în care vor fi susținute.

²⁶ Cercuri științifice, concursuri profesionale etc.

²⁷ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.