

FIȘA DISCIPLINEI

Anul universitar 2024 - 2025

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Lucian Blaga din Sibiu
1.2. Facultatea	Facultatea de Științe
1.3. Departament	Departamentul de Matematică și Informatică
1.4. Domeniul de studiu	Informatică
1.5. Ciclul de studii ¹	Licență
1.6. Specializarea	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Fundamentele Sistemelor Concurente	Cod	FSTI.MAI.INF.L.SA.6. 2020.E-5.5
2.2. Titular activități de curs	Conf. univ. dr. Florin Stoica		
2.3. Titular activități practice	Lect. univ. dr. Ralf Fabian		
2.4. An de studiu ²	3	2.5. Semestrul ³	6
2.6. Tipul de evaluare ⁴			E
2.7. Regimul disciplinei ⁵	A	2.8. Categoria formativă a disciplinei ⁶	S

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
24	-	24	-	-	48
Distribuția fondului de timp pentru studiu individual⁸					Nr. ore
Studiul după manual, suport de curs, bibliografie și notițe					29
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					9
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					30
Tutoriat ⁹					7
Examinări ¹⁰					2
3.3. Total ore alocate studiului individual¹¹ (NOS_{sem})					77
3.4. Total ore din Planul de învățământ (NOAD_{sem})					48
3.5. Total ore pe semestru¹² (NOAD_{sem} + NOS_{sem})					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	Algoritmi fundamentali, Fundamentele programării
4.2. Competențe	

5. Condiții (acolo unde este cazul)

5.1. De desfășurare a cursului ¹⁵	Sală de curs, dotată cu tablă, calculator, videoproiector și software specific (Adobe Reader, Power Point), conectare la Internet, classroom aferent disciplinei, meet (pentru desfășurarea consultațiilor, discuțiilor, sau o desfășurare online a cursului în cazul unor condiții speciale).
5.2. De desfășurare a activităților practice (lab/sem/pr/alte) ¹⁶	Sală de laborator, dotată cu tablă, calculatoare, videoproiector și software specific (Adobe Reader, Eclipse, Visual Studio), conectare la Internet, classroom aferent laboratorului disciplinei, meet (pentru desfășurarea consultațiilor, discuțiilor, sau o desfășurare online a laboratorului în cazul unor condiții speciale).

6. Competențe specifice acumulate¹⁷

Număr de credite alocat disciplinei ¹⁸		5	Repartizare credite pe competențe ¹⁹
6.1. Competențe profesionale	CP1	Potențialul de a valorifica la nivel de înțelegere competențele dobândite prin această disciplină.	4
	CP2	Abilitatea transpunere a conceptelor teoretice și a tehnologiilor fundamentale de programare concurentă în implementări practice.	
	CP3	Explicarea conceptelor și noțiunilor însușite.	
	CP4	Folosirea deprinderilor dobândite în punerea și rezolvarea de probleme practice.	
	CP5	Aptitudini de prezentare și promovare a propriilor rezultate obținute.	
	CP6	Utilizarea tehnicii de calcul pentru rezolvarea de probleme practice.	
	CP7	Studentul să dezvolte și implementeze software bazat pe biblioteci standard pentru dezvoltarea de aplicații concurente (POSIX threads, clasele JAVA, API Windows).	
	CP8	Însușirea aspectelor teoretice privind procesele și thread-urile, precum și mecanismele de sincronizare și coordonare a acestora.	
	CP9	Exploatarea facilităților de lucru cu procese și thread-uri oferite de către sistemele de operare Unix (Linux), Windows și de către platforma Java	
6.2. Competențe transversale	CT1	Dezvoltarea atitudinii pozitive față de muncă și responsabilitate pentru propria pregătire profesională	1
	CT2	Dezvoltarea spiritului de munca în echipă	
	CT3	Capacitatea de utilizare a mijloacelor moderne de documentare	

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

7.1. Obiectivul general	<ul style="list-style-type: none"> – Înțelegerea conceptelor și a tehnologiilor fundamentale de programare concurentă. – Cunoașterea tehnicilor de programare concurentă pe platforme multiple: UNIX, Windows, Java – Formarea deprinderilor necesare utilizării bibliotecilor standard pentru
-------------------------	---

	<p>dezvoltarea de aplicații concurente (POSIX threads, clasele JAVA, API Windows).</p> <ul style="list-style-type: none"> – Manifestarea interesului pentru cunoașterea rezultatelor teoretice și aplicative recente în domeniul programării concurente. – Adaptare la lucrul colaborativ, în echipă, pentru rezolvarea unor probleme din diverse domenii, prin implicarea în proiecte informatice în care se utilizează programarea concurentă
7.2. Obiectivele specifice	<ul style="list-style-type: none"> – Dobândirea cunoștințelor și aptitudinilor pentru aplicarea conceptelor teoretice de la curs în rezolvarea unor probleme concrete. – Implementarea unor algoritmi care abordează teme specifice programării concurente: excludere mutuală, resurse critice, secțiuni critice, sincronizarea accesului la resurse. – Evaluarea experimentală a performanțelor algoritmilor cu multiple fire de execuție executate în mod concurent. – Exprimarea unui mod de gândire creativ în formularea unei soluții bazată pe programare concurentă pentru probleme cu grad mediu de dificultate.

8. Conținuturi

8.1. Curs ²⁰	Metode de predare ²¹	Nr. ore
Curs 1 Nivelele prelucrărilor concurente: Procesări paralele și clasificarea Flynn. Granularități ale paralelismului / concurenței. Tehnica pipeline. Procesoare vectoriale și sisteme cluster.	Expunerea sistematică a cunoștințelor (deductivă, inductivă și formalizată, expuneri la tablă/ în meet); Conversația frontală; Conversație individuală; Conversația euristică; Problematizare; Studii de caz; Design de proiecte complexe; Modelarea și paralelizare cu fenomene general cunoscute; Învățarea prin descoperire. Discuții și explicații pe proiecte complexe	2
Curs 2 Nivelele prelucrărilor concurente: Paralelism și concurența la nivelul sistemului de operare. Evaluare multiprocesor a expresiilor complexe. Reorganizarea succesiunilor de atribuire. Paralelizare la nivel de cicluri for.		2
Curs 3 Concepte abstracte utilizate în descrierea concurenței: Paradigme de programare nesecvențială. Relația procese - thread-uri. Scheme de specificare a programelor concurente. Situații de excepție generate de concurență.		2
Curs 4 Concepte abstracte utilizate în descrierea concurenței: Mecanisme de control al concurenței, comunicare și sincronizare. Mecanisme de control asincron sau parțial sincron. Probleme specifice care se rezolvă cu ajutorul concurenței.		2
Curs 5 Programare concurentă la nivel de proces. Procese Unix, Windows, Java.		2
Curs 6 Programare concurentă la nivel de proces. Comunicarea prin pipe între procese. Comunicarea între procese folosind mecanismul de memorie partajată.		2
Curs 7 Programare concurentă la nivel de proces. Sincronizarea proceselor folosind semafoare. Comunicarea prin cozi de mesaje.		2
Curs 8 Programare concurentă la nivel de thread-uri. Caracteristici generale. Exemple de probleme rezolvabile prin thread-uri.		2
Curs 9 Programare concurentă la nivel de thread-uri. Thread-uri pe platforme Unix: Posix și Solaris. Thread-uri pe platforme Microsoft: Windows.		2
Curs 10 Programare concurentă la nivel de thread-uri. Thread-uri Java.		2
Curs 11 Aplicații concurente complexe. Scheme de proiectare a programelor concurente.		2
Curs 12 Aplicații concurente complexe. Utilizări combinate: thread-uri, procese Unix, semnale. Evaluarea unor performanțe ale programelor cu thread-uri.		2
Total ore curs:		24

8.2. Activități practice (8.2.a. Seminar ²² / 8.2.b. Laborator ²³ / 8.2.c. Proiect ²⁴ / 8.2.d. Alte act.practice ²⁵)	Metode de predare	Nr. ore
Act.1 Procese Windows. Apeluri sistem pentru managementul proceselor Windows.	Conversația frontală; Conversație individuală; Problematizare; Studii de caz; Design de proiecte complexe; Modelarea gândirii algoritmice prin exemplificare și paralelizare cu fenomene general cunoscute.	2
Act.2 Fire de execuție Windows. Apeluri sistem pentru managementul firelor de execuție Windows.		2
Act.3 Procese Linux. Apeluri sistem pentru managementul proceselor Linux.		2
Act.4 Fire de execuție Linux. Apeluri sistem pentru managementul firelor de execuție Linux.		2
Act.5 Mecanisme de comunicare inter-procese. Comunicația inter-procese prin pipe-uri. Comunicarea între procese folosind mecanismul de memorie partajată.		2
Act.6 Mecanisme de comunicare inter-procese. Comunicarea prin cozi de mesaje.		2
Act.7 Mecanisme de sincronizare (mutex-uri, variabile de condiție, semnale). Aplicație vizuală multi-threading Windows.		2
Act.8-9 Utilizarea thread-urilor în appleturi și servlet-uri Java		4
Act.10 Server Java concurent pentru chat.		2
Act.11 Client FTP noninteractiv		2
Act.12 Testare finală/evaluare finală, prezentare proiecte		2
Total ore seminar/laborator		24

9. Bibliografie

9.1. Referințe bibliografice recomandate	Michel Charpentier, Functional and Concurrent Programming: Core Concepts and Features, Addison-Wesley Professional, ISBN 978-0137466542, 2022 Stoica F., Cacovean L., Programare API, Ed. Psihomedica, 2009 F. M. Boian, C. Ferdean, R. Boian, R. Dragoș, Programare concurentă pe platforme Unix, Windows, Java, ISBN 973-650-072-1, Editura Albastră - grupul Microinformatica, Cluj, 2002.
9.2. Referințe bibliografice suplimentare	Stoica F., Sisteme de operare, Ed. Universității „Lucian Blaga”, 2007.

10. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului²⁶

- Prin utilizarea mecanismelor și tehnicilor de programare concurentă se dezvoltă aplicații software eficiente, care exploatează intensiv resursele de calcul puse la dispoziție.
Coroborarea conținutului disciplinei cu așteptările angajatorilor se realizează prin contacte periodice cu aceștia în vederea analizării și identificării soluțiilor optime pentru rezolvarea problemelor care se bazează pe arhitecturi multithreading.

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare		11.3 Pondere din nota finală	Obs. ²⁷
11.4a Examen / Colocviu	• Cunoștințe teoretice și practice însușite (cantitatea, corectitudinea, acuratețea)	Teste pe parcurs ²⁸ :	-	50%	CEF
		Teme de casă:	-		
		Alte activități ²⁹ :	-		
		Evaluare finală:	100%		
11.4b Seminar	• Frecvența/relevanța intervențiilor sau răspunsurilor	Evidența intervențiilor, portofoliu de lucrări (referate, sinteze științifice)		-	-
11.4c Laborator	• Cunoașterea aparatului, a modului de utilizare a	Activități aplicative	20%	50%	nCPE



	instrumentelor specifice; evaluarea unor instrumente sau realizări, prelucrarea și interpretarea unor rezultate				
11.4d Proiect	<ul style="list-style-type: none"> Calitatea proiectului realizat, corectitudinea documentației proiectului, justificarea soluțiilor alese 	<ul style="list-style-type: none"> Autoevaluarea, prezentarea și/sau susținerea proiectului Evaluarea critică a unui proiect 	80%		
11.5 Standard minim de performanță ³⁰ Înțelegerea conceptelor și a tehnologiilor fundamentale de programare concurentă Cunoașterea tehnicilor de programare concurentă pe platforme multiple: Windows, Java Pentru intrarea în examen, sunt necesare minim 8 prezențe la activitățile de laborator.					

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: |_0_|_5_| / |_0_|_9_| / |_2_|_0_|_2_|_4_|

Data avizării în Departament: |_1_|_7_| / |_0_|_9_| / |_2_|_0_|_2_|_4_|

	Grad didactic, titlul, prenume, numele	Semnătura
Titular disciplină	Conf. univ. dr. Florin Stoica	
Responsabil program de studii	Prof. univ. dr. Dana Simian	
Director Departament	Prof. univ. dr. Mugur Acu	

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.d.e.)

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)

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

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 etc.

²³ Demonstrație practică, exercițiu, experiment etc.

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

²⁵ Alte tipuri de activități practice specifice

²⁶ Legătura cu alte discipline, utilitatea disciplinei pe piața muncii

²⁷ CPE – condiționează participarea la examen; nCPE – nu condiționează participarea la examen; CEF - condiționează evaluarea finală; N/A – nu se aplică

²⁸ Se va preciza numărul de teste și săptămânile în care vor fi susținute.

²⁹ Cercuri științifice, concursuri profesionale etc.

³⁰ Se particularizează la specificul disciplinei standardul minim de performanță din grila de competențe a programului de studii, dacă este cazul.