

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. Ciclu de studii ¹	Master
1.6. Specializarea	Sisteme și Tehnologii Informatică Avansate

2. Date despre disciplină

2.1. Denumirea disciplinei	Arhitecturi de calcul paralel și distribuit	Cod	FSTI.MAI.STIA.M.SA .3.1020.E-6.21
2.2. Titular activități de curs	Conf. univ. dr. Florin Stoica		
2.3. Titular activități practice	Lect. univ. dr. Constantin Maniu		
2.4. An de studiu ²	2	2.5. Semestrul ³	3
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
1	-	2	-	-	3
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
14	-	28	-	-	42
Distribuția fondului de timp pentru studiu individual⁸					Nr. ore
Studiul după manual, suport de curs, bibliografie și notițe					38
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					18
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					40
Tutoriat ⁹					10
Examinări ¹⁰					2
3.3. Total ore alocate studiului individual¹¹ (NOSI_{sem})					108
3.4. Total ore din Planul de învățământ (NOAD_{sem})					42
3.5. Total ore pe semestru¹² (NOAD_{sem} + NOSI_{sem})					150
3.6. Nr ore / ECTS					25
3.7. Număr de credite¹³					6

4. Precondiții (acolo unde este cazul)

4.1. Discipline necesar a fi promovate anterior (de curriculum) ¹⁴	Fundamentele programării, Sisteme de Operare
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 cu procesoare multi-core, videoproiector și software specific (Adobe Reader), 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	6	Repartizare credite pe competențe ¹⁹
6.1. Competențe profesionale	CP1	Înțelegerea unor arhitecturi paralele avansate de procesare a informației precum și a metodologiilor de cercetare ale acestora		5
	CP2	Explicarea și interpretarea calitativă a rezultatelor cantitative obținute prin simulări complexe. Înțelegerea acțiunii diverșilor parametri asupra performanței sistemelor avansate de calcul. Stăpânirea complexității proiectelor arhitecturale.		
	CP3	Înțelegerea aprofundată a dependenței dintre performanța aplicațiilor software și caracteristicile arhitecturii hardware, capacitatea de a interpreta rezultatele obținute.		
	CP4	Dezvoltarea aptitudinilor practice de lucru cu sisteme de calcul avansate, medii de dezvoltare, compilatoare, debuggere, benchmark-uri etc., aferente unor standarde de programare paralelă (OpenMP, MPI), în vederea proiectării-dezvoltării de aplicații hardware-software integrate și evaluării performanțelor acestora.		
6.2. Competențe transversale	CT1	capacitatea de utilizare a mijloacelor moderne de documentare și de simulare/evaluare a arhitecturilor complexe;		1
	CT2	crearea unui limbaj tehnic adecvat analizelor și dezvoltărilor experimentale în domeniul calculului paralel;		
	CT3	capacitatea studenților de a lucra în echipe de câte 2-4 membri, în vederea efectuării unor aplicații relativ complexe.		

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

7.1. Obiectivul general	<ul style="list-style-type: none"> – Studiul unor arhitecturi moderne de arhitecturi paralele, folosite în calculul de mare performanță. – Studiul unor tehnici de paralelizare a algoritmilor. – Prezentarea conceptelor care stau la baza implementării paralelismului: fire de execuție, procese, variabile partajate, mutexuri, semafoare, transfer de mesaje, etc.
7.2. Obiectivele specifice	<ul style="list-style-type: none"> – Construirea de aplicații paralele utilizând standardul de multiprocesare OpenMP – Dezvoltarea de aplicații paralele utilizând biblioteca MPI (Message Passing

Interface)

8. Conținuturi

8.1. Curs ²⁰	Metode de predare ²¹	Nr. ore
Curs 1 Arhitectura calculatoarelor paralele. Tehnici de creștere a performanțelor computaționale. Introducerea paralelismului în interiorul procesorului. Tehnica pipeline. Prelucrarea vectorială.	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	1
Curs 2 Calculatoare paralele. Clasificarea calculatoarelor în funcție de mecanismul de control (SISD, MISD, SIMD, MIMD). Clasificarea calculatoarelor paralele după organizarea spațiului de adresă al memoriei. Sisteme paralele și distribuite.		1
Curs 3 Sisteme cluster. Sisteme grid. Sisteme Cloud Computing.		1
Curs 4 Rețele de interconectare din calculatoarele paralele (dinamice: crossbar, magistrala, multietaj, statice: stea, lanț, inel, grilă, tor, arbore, hipercub). Rutarea datelor. Rețele de interconectare în clustere de calculatoare (Gigabit Ethernet, Infiniband, Myrinet).		1
Curs 5 Analiza și proiectarea algoritmilor paraleli. Dependente între task-uri. Condiții de paralelism. Etapele de dezvoltare a algoritmilor paraleli. Reprezentarea algoritmilor. Dimensiunea problemei. Granularitatea algoritmului paralel. Comunicația între task-urile unui algoritm paralel. Gradul de paralelism al unui algoritm.		1
Curs 6 Algoritmii SAXPY. Gruparea task-urilor pentru algoritmul SAXPY. Paralelizarea algoritmului. Algoritmii de adunare a două matrice în paralel.		1
Curs 7 Estimarea și analiza performanțelor algoritmilor paraleli. Estimarea timpului de execuție. Complexitatea algoritmilor. Accelerarea paralelă. Eficiența și costul paralel.		1
Curs 8 Legea lui Amdahl. Modelul teoretic PRAM de calculator paralel. Algoritmi PRAM.		1
Curs 9 Programarea algoritmilor paraleli. Programare concurentă. Biblioteci, limbaje și compilatoare paralele. Procese Unix. Procese Windows. Memorie partajată. Thread-uri POSIX.		1
Curs 10 Programare paralelă prin variabile partajate. Comunicarea și sincronizarea între procese. Excluderea mutuală. Mutexuri. Semafoare. Monitoare. Variabile de condiție. Multiprocesarea buclelor.		1
Curs 11-12 Tehnologia OpenMP. Directive OpenMP. Paralelizarea automată a programelor. Compilatoare cu paralelizare automată.		1 1
Curs 13-14 Programare paralelă prin transfer de mesaje. Primitive de transfer blocant/neblocant de mesaje. Comunicația client/server. Biblioteca MPI.		2 2
Total ore curs:		14

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.1-2 Introducere în sisteme Linux/Unix. Comenzi referitoare la gestiunea proceselor. Execuția fișierelor executabile în imaginea unui proces.	Conversația frontală; Conversație individuală; Problematizare; Studii de caz; Design de proiecte complexe; Modelarea gândirii algoritmice prin exemplificare și paralelizare	4
Act.3-4 Sincronizarea între procese prin semafoare, mutex-uri, variabile de condiție.		4
Act.5-6 Comunicația între procese prin segmente de memorie partajată.		4
Act.7-8 Fire de execuție POSIX.		4
Act.9-10 Biblioteca OpenMP.		4

Act.11-12 Biblioteca MPI.	cu fenomene general cunoscute.	4
Act.13-14 Dezvoltare/Prezentare proiecte.		4
Total ore seminar/laborator		28

9. Bibliografie

9.1. Referințe bibliografice recomandate	Peter Pacheco, Matthew Malensek, An Introduction to Parallel Programming, Second Edition, Elsevier, 2022
	Grigoraș D., "Calcul Paralel. De la Sisteme la programarea aplicațiilor", Computer Libris Agora, Cluj, 2000
	Boian F.M. "Programarea distribuită în Internet", Ed. Albastră, Cluj-Napoca, 1999
	Stoica F., Cacovean L., Programare API, Ed. Promedia, 2009
9.2. Referințe bibliografice suplimentare	

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

<ul style="list-style-type: none"> – Disciplina Arhitecturi de calcul paralel și distribuit își găsește aplicabilitate în toate domeniile de activitate care implică efectuarea de calcule de mare performanță: predicții meteorologice, modele economice, inteligență artificială și automatizări, explorări seismice, inginerie genetică, aplicații medicale, etc. Firmele sau instituțiile de cercetare care desfășoară activități relaționate domeniilor amintite pot utiliza tehnici de calcul paralel și tehnologii distribuite în cadrul unor infrastructuri moderne, bazate pe sisteme multiprocesor. – Coroborarea conținuturilor disciplinei se realizează prin contacte periodice cu reprezentanții companiilor de profil în vederea analizei problemelor specifice care necesită calcule intensive / arhitecturi paralele sau distribuite.
--

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%	nCPE 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 aparaturii, a modului de utilizare a instrumentelor specifice; evaluarea unor instrumente sau realizări, prelucrarea și interpretarea unor rezultate	Activități aplicative	20%	50%	nCPE, CEF
11.4d Proiect	• Calitatea proiectului realizat, corectitudinea documentației proiectului, justificarea soluțiilor alese	• Autoevaluarea, prezentarea și/sau susținerea proiectului • Evaluarea critică a unui proiect	80%		
11.5 Standard minim de performanță ³⁰					
<ul style="list-style-type: none"> – Capacitatea de a dezvolta aplicații paralele într-un limbaj de programare – Însușirea conceptelor care stau la baza implementării paralelismului: fire de execuție, 					



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.