FIŞA DISCIPLINEI

1. Date despre program

Instituţia de învăţământ superior	Universitatea „Ştefan cel Mare” Suceava
Facultatea	Inginerie Electrică şi Ştiinţa Calculatoarelor
Departamentul	Calculatoare
Domeniul de studii	Calculatoare şi Tehnologia Informaţiei
Ciclul de studii	Master de cercetare
Programul de studii	Știința și Ingineria Calculatoarelor (SIC)

2. Date despre disciplină

Denumirea disciplinei	**ALGORITMI PARALELI AVANSAȚI**				
Anul de studiu	**I**	Semestrul	**2**	Tipul de evaluare	**E**
Regimul disciplinei	Categoria formativă a disciplinei DF - fundamentală, DS - de specializare, DC – complementară				**DS**
	Categoria de opționalitate a disciplinei: DOB – obligatorie, DOP – opțională, DFA - facultativă				**DOB**

3. Timpul total estimat (ore alocate activităţilor didactice)

I a) Număr de ore, pe săptămână	2	Curs	**1**	Seminar	**0**	Laborator/lucrări practice	**1**	Proiect	**0**
I.b) Totalul de ore (pe semestru) din planul de învăţământ	28	Curs	**14**	Seminar	**0**	Laborator/lucrări practice	**14**	Proiect	**0**

Distribuția fondului de timp pe semestru	ore
II. Studiu individual	**94**
III. Examinări	**3**
IV. Alte activități (precizați):	

Total ore studiu individual (II+III)	**97**
Total ore pe semestru (I.b+II+III+IV)	**125**
Numărul de credite	**5**

4. Competențe specifice acumulate

Competențe transversale	CT1. Lucrează în echipe;
Competențe profesionale	CP5. Asigură managementul de proiect; CP6. Aplică politici de securitate informatică; CP7. Administrează datele; CP8. Dezvoltă aplicații de procesare de date;

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul / absolventul identifică, conştientizează critic, sumarizează concepte și metode avansate privitoare la algoritmi pentru aplicații de înaltă performanță și modul lor de aplicare în rezolvarea inovativă a unor probleme concrete.	Studentul / absolventul cercetează și proiectează aplicații care folosesc procesarea paralelă a datelor, gestionând probleme de sincronizare, partiționare și comunicare între procese și utilizează algoritmi paraleli avansați.	Studentul / absolventul își poate asuma responsabilitatea pentru a contribui la cunoştinţele şi practicile profesionale.

6. **Obiectivele disciplinei** (reieşind din grila competenţelor specifice acumulate)

Obiectivul general al disciplinei	Se urmărește înțelegerea principiilor care stau la baza proiectării, analizei și implementării algoritmilor paraleli pentru sisteme cu memorie distribuită, precum și a modului de funcționare a unui cluster de calcul de înaltă performanță (HPC). Se urmărește formarea capacității de a dezvolta, folosind biblioteca MPI în limbajele C/C++, aplicații paralele corecte, eficiente și scalabile. Însușirea algoritmilor paraleli avansați și a tehnicilor fundamentale de evaluare a performanței în acest domeniu.
Obiectivele specifice	- cunoașterea modelelor de calcul paralel și a măsurilor de performanță (accelerare, eficiență, cost, scalabilitate); - aplicarea unei metodologii sistematice de proiectare a algoritmilor paraleli (partiționare, comunicare, aglomerare, mapare); - utilizarea bibliotecii MPI pentru comunicarea punct-la-punct și colectivă, tipuri de date derivate și topologii virtuale; - studiul algoritmilor paraleli pentru operații cu matrice, sisteme de ecuații liniare, sortare și grafuri; - implementarea, testarea și rularea programelor paralele pe clusterul universității; - validarea și interpretarea rezultatelor obținute prin implementarea algoritmilor paraleli avansați.

7. **Conținutul predării și învățării**

Curs	Nr. ore	Metode de predare	Observaţii
1. Introducere în calculul paralel. Proiectarea și analiza algoritmilor paraleli 1.1. Modele de calcul paralel (PRAM, transmitere de mesaje, BSP); taxonomia Flynn; sisteme cu memorie partajată și distribuită; clusterul HPC al USV; 1.2. Măsuri de performanță: accelerare (speedup), eficiență, cost; legile lui Amdahl și Gustafson; scalabilitate tare și slabă; izoeficiență; 1.3. Metodologia PCAM (partiționare, comunicare, aglomerare, mapare); tehnici de descompunere; echilibrarea încărcării; modelul master–worker.	3	expunerea, prelegerea-dezbatere, demonstraţia	
2. Programarea paralelă cu biblioteca MPI în C/C++ 2.1. Modelul SPMD; comunicatori și ranguri; comunicare punct-la-punct blocantă și neblocantă; evitarea blocajelor (deadlock); modelul de cost latență–lățime de bandă; 2.2. Operații de comunicare colectivă (broadcast, scatter/gather, reduce/allreduce, all-to-all, scan) și algoritmii care le implementează (arbore binomial, inel, dublare recursivă); 2.3. Tipuri de date derivate; comunicatori derivați; topologii virtuale carteziene și maparea proceselor; 2.4. Măsurarea timpului de execuție și a performanței programelor MPI.	4	expunerea, prelegerea-dezbatere, demonstraţia	
3. Algoritmi paraleli pentru matrice și sisteme de ecuații liniare 3.1. Înmulțirea matrice–vector: partiționare pe linii, pe coloane și 2D; 3.2. Înmulțirea matrice–matrice: algoritmul cu blocuri, algoritmul lui Cannon, SUMMA; 3.3. Eliminarea Gaussiană paralelă: distribuție ciclică, pivotare parțială, pipeline;	4	expunerea, prelegerea-dezbatere, demonstraţia	

3.4. Metode iterative (Jacobi, Gauss–Seidel roșu–negru) cu schimb de frontiere (halo exchange).			
4. Algoritmi paraleli de sortare 4.1. Operația compare–split; sortarea par–impar prin transpoziție (odd-even transposition); 4.2. Sortarea bitonică; Quicksort paralel; Sample sort.	2	expunerea, prelegerea-dezbatere, demonstrația	
5. Algoritmi paraleli pe grafuri 5.1. Partiționarea grafurilor; parcurgerea în lățime (BFS) paralelă; 5.2. Arbori de acoperire minimi (Prim, Borůvka); drumuri minime (Dijkstra, Floyd–Warshall); 5.3. Mulțimi independente maximale (algoritmul lui Luby).	1	expunerea, prelegerea-dezbatere, demonstrația	

Bibliografie minimală recomandată

1. P. Pacheco, M. Malensek, An Introduction to Parallel Programming, 2nd ed., Morgan Kaufmann, 2022
2. A. Grama, A. Gupta, G. Karypis, V. Kumar, Introduction to Parallel Computing, 2nd ed., Addison-Wesley, 2003
3. P. Sanders, K. Mehlhorn, M. Dietzfelbinger, R. Dementiev, Sequential and Parallel Algorithms and Data Structures: The Basic Toolbox, Springer, 2019, ISBN 978-3-030-25208-3
4. W. Gropp, E. Lusk, A. Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, 3rd ed., MIT Press, 2014
5. T. Rauber, G. Rünger, Parallel Programming for Multicore and Cluster Systems, 3rd ed., Springer, 2023
6. W. Hwu, D. Kirk, I. El Hajj, Programming Massively Parallel Processors, 4th ed., Morgan Kaufmann, 2022
7. Petcu D., "Procesare paralelă", Editura Eubeea, Timișoara, 2001, ISBN 973-9479-48-0
8. Petcu D., Negru V., "Procesare distribuită", Editura Universității de Vest, Timișoara, 2002, ISBN 973-85552-8-0
9. Cristea V., "Algoritmi de prelucrare paralelă", Ed. Matrix Rom, București, 2005
10. Croitoru C., "Introducere în proiectarea algoritmilor paraleli", Ed. Matrix Rom, București, 2004
11. Grigoraş D., "Calcul paralel. De la sisteme la programarea aplicaţiilor", Computer Libris Agora, Cluj, 2000

Alte surse

1. Message Passing Interface Forum, MPI: A Message-Passing Interface Standard, Version 4.1, 2023, https://www.mpi-forum.org/docs/
2. I. Foster, Designing and Building Parallel Programs, Addison-Wesley, 1995, https://www.mcs.anl.gov/~itf/dbpp/
3. V. Eijkhout, Parallel Programming in MPI and OpenMP, 2022, https://theartofhpc.com/

Aplicaţii (Seminar / laborator / lucrări practice / proiect)	**Nr. ore**	**Metode de predare**	**Observaţii**
1. Mediul de lucru pe clusterul USV: compilarea (mpicc/mpicxx) și lansarea programelor MPI (mpirun, Slurm). Primul program MPI în C/C++; măsurarea timpului cu MPI_Wtime	2	lucrări practice, experimentul	
2. Comunicarea punct-la-punct: inel de procese, ping-pong (latență și lățime de bandă). Calculul lui PI prin integrare numerică	2	lucrări practice, experimentul, verificare/testare	
3. Operații colective MPI: calculul lui PI prin metoda Monte Carlo; ciurul lui Eratostene paralel (numere prime)	2	lucrări practice, experimentul	
4. Operații cu matrice: înmulțirea matrice–vector; înmulțirea a două matrice (algoritmul lui Cannon pe topologie carteziană)	2	lucrări practice, experimentul	
5. Sisteme de ecuații: eliminarea Gaussiană paralelă; metoda Jacobi cu schimb de frontiere (halo exchange)	2	lucrări practice, experimentul	
6. Sortare paralelă: sortarea par–impar prin transpoziție (odd-even transposition); Sample sort	2	lucrări practice, experimentul	
7. Algoritmi paraleli pe grafuri (Floyd–Warshall, BFS). Evaluarea performanței: accelerare, eficiență, scalabilitate. Susținerea temelor de casă	2	lucrări practice, experimentul	
Temele de casă constau în dezvoltarea unor proiecte din domeniul calculului paralel de înaltă performanță, pe platforme de tip cluster, utilizând limbajele C/C++ și biblioteca MPI, împreună cu documentația aferentă (raport tehnic).			

Etape principale în dezvoltarea temei

- însușirea temei și analiza versiunii secvențiale;
- proiectarea algoritmului paralel (partiționare, comunicare, mapare);
- programare; inserarea de comentarii în codul sursă;
- punerea la punct și testarea cu date impuse;
- măsurarea timpilor de execuție pentru diferite numere de procese și dimensiuni ale datelor;
- calculul accelerării și eficienței; validarea și interpretarea rezultatelor obținute.

În cadrul proiectelor se urmărește ca produsele-program realizate de studenți să fie implementate și evaluate experimental pe structuri paralele de calcul. Exemple de teme de proiect propuse:

- Înmulțirea matrice–vector pe structura de calcul paralelă, cu analiza scalabilității
- Înmulțirea matrice–matrice (algoritmul lui Cannon / SUMMA) pe structura de calcul paralelă
- Rezolvarea unui sistem de ecuații liniare prin eliminare Gaussiană paralelă
- Metoda Jacobi pentru ecuația căldurii pe o grilă 2D (halo exchange)
- Transformata Fourier Rapidă (FFT) implementată pe o structură de calcul paralelă
- Sortare paralelă: sortare bitonică, Sample sort, Quicksort paralel – studiu comparativ
- Algoritmi paraleli pe grafuri: BFS, Floyd–Warshall, arbori de acoperire minimi
- Algoritmi de căutare paralelă; problema celor N corpuri

Bibliografie minimală recomandată

1. M. J. Quinn, Parallel Programming in C with MPI and OpenMP, McGraw-Hill, 2004, ISBN 978-0-07-123265-4
2. P. Pacheco, M. Malensek, An Introduction to Parallel Programming, 2nd ed., Morgan Kaufmann, 2022
3. W. Gropp, E. Lusk, A. Skjellum, Using MPI, 3rd ed., MIT Press, 2014
4. A. Grama, A. Gupta, G. Karypis, V. Kumar, Introduction to Parallel Computing, 2nd ed., Addison-Wesley, 2003
5. Petcu D., "Algoritmi paraleli", Tipografia Universității din Timișoara, 1994
6. Cristea V., "Algoritmi de prelucrare paralelă", Ed. Matrix Rom, București, 2005
7. Croitoru C., "Introducere în proiectarea algoritmilor paraleli", Ed. Matrix Rom, București, 2004

Alte surse

1. V. Eijkhout, Parallel Programming in MPI and OpenMP, 2022, https://theartofhpc.com/
2. Documentația Open MPI: https://docs.open-mpi.org/
3. Documentația MPICH: https://www.mpich.org/documentation/

Observații:

- *La cererea studentului, activitatea pe parcurs poate fi echivalată prin proiecte, pregătirea și participarea la concursuri profesionale sau alte tipuri de activități specifice, cu aprobarea cadrului didactic care conduce lucrările practice;*
- *Studenții care au mai mult de 50% absențe la activitatea practică pot recupera această activitate cu aprobarea cadrului didactic care conduce lucrările practice.*

8. **Evaluare**

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Evaluare finală	Criterii generale de evaluare (corectitudinea și completitudinea cunoştinţelor, coerenţa logică, fluenţa în redactare) Criterii specifice disciplinei (proiectarea și analiza algoritmilor paraleli, utilizarea corectă a bibliotecii MPI și a terminologiei de specialitate)	Evaluare sumativă prin probă scrisă și probă practică pe calculator (*subiecte teoretice, probleme de proiectare și analiză a algoritmilor paraleli, program MPI funcțional*)	50
Evaluare pe parcurs	Criterii generale de evaluare (corectitudinea cunoştinţelor, coerenţa logică, fluenţa în redactare) Criterii specifice disciplinei (funcționalitatea și corectitudinea programelor MPI, corectitudinea măsurătorilor de performanță și interpretarea lor, calitatea documentației)	Evaluare pe parcurs prin verificarea lucrărilor de laborator și a temei de casă (*scrierea și punerea la punct a unor programe MPI funcționale pe cluster, raport tehnic și susținere*)	50

Fișa disciplinei include, dacă este cazul, elemente adaptate persoanelor cu dizabilități, în funcție de tipul și gradul acestora.

Data completării	Grad didactic, nume, prenume, semnătura titularului de curs	Grad didactic, nume, prenume, semnătura titularului de aplicație

23.09.2026	s.l. univ.dr.ing. Remus Cătălin PRODAN	s.l. univ.dr.ing. Remus Cătălin PRODAN

Data avizării	Grad didactic, nume, prenume, semnătura responsabilului de program
24.09.2026	prof. dr. ing. Stefan-Gheorghe PENTIUC

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
24.09.2026	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
25.09.2026	Prof. dr. ing. Laurențiu-Dan MILICI