

FIȘA DISCIPLINEI

1. Date despre program

Facultatea	Inginerie Electrică și Știința Calculatoarelor
Departamentul	Calculatoare
Domeniul de studii	Calculatoare și Tehnologia Informației
Ciclul de studii	Masterat
Programul de studii	Masterat Știința și Ingineria Calculatoarelor

2. Date despre disciplină

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

3. Timpul total estimat (ore alocate activităților 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 învățământ	28	Curs	14	Seminar	0	Laborator/lucrări practice	14	Proiect	0

II. Distribuția fondului de timp pe semestru	ore
II.a) Studiu individual	97
II.b) Tutoriat (pentru ID)	0
III. Examinări	0
IV. Alte activități:	0

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

4. Competențe specifice acumulate

Competențe profesionale	CP5. Asigura managementul de proiect; CP6. Aplica politici de securitate informatica; CP7. Administreaza datele; CP8. Dezvolta aplicatii de procesare de date;
Competențe transversale	CT1. lucreaza în echipe;

5. Rezultatele învățării

Cunoștințe	Aptitudini	Responsabilitate și autonomie
Studentul / absolventul identifică, conștientizează 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 si utilizează algoritmi paraleli avansați.	Studentul / absolventul își poate asuma responsabilitatea pentru a contribui la cunoștințele și practicile profesionale.

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

Obiectivul general al disciplinei	Se urmărește înțelegerea ideilor care stau la baza metodelor de calcul paralel, a principiilor de funcționare a sistemelor de calcul paralele, a configurării unui cluster, prin conectarea mai multor platforme de calcul. Se urmărește și modul în care paralelizarea operațiilor a contribuit istoric la dezvoltarea microprocesoarelor, supercalculatoarelor și chiar a calculatoarelor personale. Înșușirea notiunilor fundamentale în domeniul algoritmilor de calcul paralel, respectiv asimilarea algoritmilor avansați și tehnicilor fundamentale în acest domeniu
	- introducerea noțiunilor fundamentale specifice procesării paralele; - descrierea tehnologiilor moderne utilizate în calculului de înaltă performanță - HPC; - prezentarea bibliotecii MPI, utilizată în scrierea și experimentarea algoritmilor de calcul paralel; - analiza problemelor ce se pretează la implementări pe structuri paralele de calcul. - studiul algoritmilor utilizați în calculului de înaltă performanță; - validarea și interpretarea rezultatelor obținute prin implementarea algoritmilor avansați de calcul paralel;

7. Conținutul predării și învățării

Curs	Nr. ore	Metode de predare	Observații
1. Introducere în calculul paralel 1.1. Scopul cursului; 1.2. Concepte și noțiuni. 1.3. Modele de arhitecturi paralele	1	expunerea, prelegerea-dezbateri, demonstrația	
2. Arhitecturi de calcul paralel 2.1. Clustere beowulf. Structura hardware și software. Performanță și costuri; 2.2. RedPower HPC, clusterul de la USV. Arhitectura hardware și software;	2	expunerea, prelegerea-dezbateri, demonstrația	
3. Comunicarea în mașini paralele, biblioteca MPI 3.1. Modele și costuri; 3.2. Tipuri de date; 3.3. Operații de comunicare colectivă 3.4. Topologii de procese; 3.5. Maparea topologiilor 3.6. Algoritmi simpli utilizând MPI (calcul integral definit, număr PI, etc)	4	expunerea, prelegerea-dezbateri, demonstrația	
4. Exemple de algoritmi paraleli. Prelucrarea masivelor de date 4.1. Calculul lui Pi 4.2. Înmulțirea vectorilor și al matricelor; 4.3. Sisteme de N ecuații liniare; 4.4. Eliminarea Gaussiana.	5	expunerea, prelegerea-dezbateri, demonstrația	
5. Exemple de algoritmi paraleli: Grafuri 5.1. Arbori de acoperire minimi; 5.2. Drumuri minime; 5.3. Maximal Independent Set	2	expunerea, prelegerea-dezbateri, demonstrația	
Bibliografie minimală recomandată			
1. Sanders, P., Mehlhorn, K., Dietzfelbinger, M., Dementiev, R., Sequential and Parallel Algorithms and Data Structures: The Basic Toolbox, ISBN: 9783030252090, 2019; 2. P. Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2010 3. A. Grama, A. Gupta, G. Karypis, V. Kumar: Introduction to Parallel Computing, Second Edition, 2003, AddisonWesley 4. Petcu D., "Procesare paralelă", Editura Eubee, 2001, Colecția Informatica, Timișoara, ISBN 973-9479-48-0, 414 pagini. 5. Petcu D., Negru V., "Procesare distribuită", Editura Universității de Vest, Seria Alef, 2002, Timișoara, ISBN 973-85552-8-0, 420 pagini 6. Petcu D., "Algoritmi paraleli", Tipografia Universității Timișoara, 1994, 205 pagini.			

7. Dana Petcu (@Universitatea de Vest Timisoara), Procesare paralela, Ed. Eubeea Timisoara, 2001
<http://www.info.uvt.ro/~petcu>
8. T. Rauber, G. Runger, Parallel Programming for multicore and cluster systems, Springer, 2013
9. D. Kirk, Programming massively parallel processors, Morgan Kaufmann, 2013
10. Chiorean I., "Calcul paralel. Fundamente", Ed. Microinformatica, 1995
11. Cristea V., "Algoritmi de prelucrare paralelă", Ed. Matrix Rom, 2005
12. Croitoru c., "Introducere in proiectarea algoritmilor paraleli", Ed. Matrix Rom, 2004
13. Grigoraș D., "Calcul Paralel. De la Sisteme la programarea aplicațiilor", Computer Libris Agora, Cluj, 2000
14. Hockney R.W., Jesshope C.R., "Calculatoare paralele. Arhitectura, programare, algoritmi", Ed. Tehnică, 1991

Alte surse

1. <http://www-users.cs.umn.edu/~karypis/parbook/>
2. Ian Foster: Designing and Building Parallel Programs, 1995, Addison Wesley
<http://wotug.kent.ac.uk/parallel/books/addison-wesley/dbpp/>

Aplicații (Seminar / laborator / lucrări practice / proiect)	Nr. ore	Metode de predare	Observații
1. Programare C in Linux. Apeluri sistem. Procese si thread-uri	2	lucrări practice, experimentul	
2. Introducere in librăria MPI (<i>Message-Passing Interface</i>)	2	lucrări practice, experimentul, verificare/testare	
3. Algoritmi paraleli de calcul al valorii lui PI. Algoritmi de operare cu numere prime	2	lucrări practice, experimentul	
4. Algoritm de sortare. Algoritmul Odd-Even-Par. Sortarea prin interschimbare	2	lucrări practice, experimentul	
5. Operații cu matrici. Înmulțirea vector-matrice. Înmulțirea a doua matrici	2	lucrări practice, experimentul	
6. Algoritmi in prelucrarea numerica a imaginilor. Calculul Transformatei Fourier	2	lucrări practice, experimentul	
7. Sisteme de ecuații. Reducerea la forma triunghiulara. Algoritmul eliminării Gaussiene. Versiunea paralela	2	lucrări practice, experimentul	
Temele de casă consta în dezvoltarea unor proiecte, din domeniul calculului paralel de inalta performanta, pe platforme de tip cluster, utilizand limbaje de programare C/C++, impreuna cu documentatia aferenta.. Etape principale în dezvoltarea temei - însușirea temei de realizare; - proiectarea logică a produsului program; - programare; inserarea de comentarii în codul sursă - punerea la punct a produsului program; - testarea cu date impuse; - validarea si interpretarea datelor obtinute În cadrul proiectelor se urmărește ca produsele-program realizate de studenți să fie implementate pe structuri de calcul. Exemple de teme de proiect propuse: - Înmulțirea vector-matrice realizata pe structura de calcul paralela - Înmulțirea matrice-matrice realizata pe structura de calcul paralela - Rezolvarea unui sistem de ecuații pe structura de calcul paralela - Algoritmi de căutare implementați pe o structura de calcul paralela - Transformata Fourier implementata pe o structura de calcul paralela - Algoritmul Bubble-Sort implementat pe o structura de calcul paralela - Algoritmul QuickSort implementat pe o structura de calcul paralela - Alți algoritmi de sortare implementați pe structuri de calcul paralel			
Bibliografie minimală recomandată			
1. Sanders, P., Mehlhorn, K., Dietzfelbinger, M., Dementiev, R., Sequential and Parallel Algorithms and Data Structures: The Basic Toolbox, ISBN: 9783030252090, 2019; 2. Michael J. Quinn, Parallel Programming in C with Mpi and Openmp, McGraw-Hill Education, ISBN-13: 978-0071232654, 2008 3. A. Grama, A. Gupta, G. Karypis, V. Kumar: Introduction to Parallel Computing, Second Edition, 2003, AddisonWesley 4. Petcu D., "Procesare paralelă", Editura Eubeea, 2001, Colecția Informatica, Timișoara, ISBN 973-9479-48-0, 414 pagini. 5. Petcu D., "Algoritmi paraleli", Tipografia Universității Timișoara, 1994, 205 pagini. 6. Chiorean I., "Calcul paralel. Fundamente", Ed. Microinformatica, 1995 7. Cristea V., "Algoritmi de prelucrare paralelă", Ed. Matrix Rom, 2005 8. Croitoru c., "Introducere in proiectarea algoritmilor paraleli", Ed. Matrix Rom, 2004			

9. Grigoraș D., "Calcul Paralel. De la Sisteme la programarea aplicațiilor", Computer Libris Agora, Cluj, 2000
10. Hockney R.W., Jesshope C.R., "Calculatoare paralele. Arhitectura, programare, algoritmi", Ed. Tehnică, 1991
Alte surse
1. http://www-users.cs.umn.edu/~karypis/parbook/
2. Ian Foster: Designing and Building Parallel Programs, 1995, Addison Wesley http://wotug.kent.ac.uk/parallel/books/addison-wesley/dbpp/
Observatii:
- <i>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;</i>
- <i>Studentii care au mai mult de 50% absențe la activitatea practică pot recupera această activitate cu aprobarea cadrului didactic care conduce lucrările practice.</i>

8. Evaluare

Tip activitate	Criterii de evaluare	Metode de evaluare	Pondere din nota finală
Curs	Criteriile generale de evaluare (corectitudinea cunoștințelor, coerența logică, fluența în redactare) Criterii specifice disciplinei (utilizarea corectă a metalimbajului specific domeniului limbajelor de programare)	Evaluare sumativă prin examinare scrisă (<i>proba orală și proba practică pe calculator</i>)	50
Seminar	-	-	-
Laborator	Criteriile generale de evaluare (corectitudinea cunoștințelor, coerența logică, fluența în redactare) Criterii specifice disciplinei (utilizarea corectă a metalimbajului specific domeniului limbajelor de programare)	Evaluare pe parcurs prin examinare scrisă (<i>prin scrierea și punerea la punct a unui program funcțional pe calculator</i>)	50
Proiect	-	-	-

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
25.09.2025	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
25.09.2025	prof. dr. ing. Ștefan-Gheorghe PENTIUC

Data avizării în departament	Grad didactic, nume, prenume, semnătura directorului de departament
26.09.2025	Prof. dr. ing. Ovidiu-Andrei SCHIPOR

Data aprobării în consiliul facultății	Grad didactic, nume, prenume, semnătura decanului
26.09.2025	Prof. dr. ing. Laurentiu- Dan MILICI