



FIȘA DISCIPLINEI (ALGORITMI DE OPTIMIZARE IN SISTEME DISTRIBUITE)

1. Date despre program

1.1 Instituția de învățământ superior	UNIVERSITATEA „OVIDIUS” DIN CONSTANȚA
1.2 Facultatea	Facultatea de Matematica și Informatică
1.3 Departamentul	Matematica și Informatică
1.4 Domeniul de studii	Informatică
1.5 Ciclul de studii	Masterat
1.6 Programul de studii	Medii Virtuale Multi-Modale Distribuite
1.7 Anul universitar	2025-2026

2. Date despre disciplină

2.1 Denumirea disciplinei	Algoritmi de optimizare in sisteme distribuite						
2.2 Cod disciplină	FMI.MVMMD.II.2.05						
2.3 Titularul activităților de curs	Conf.univ.dr. Serban Cristina						
2.4 Titularul activităților aplicative	Conf.univ.dr. Serban Cristina						
2.5 Anul de studii	2	2.6 Semestrul	2	2.7 Tipul de evaluare	E	2.8 Regimul disciplinei	DAP/DOB

* DF – disciplină fundamentală, DS – disciplină de specializare, DC – disciplină complementară, DAP – disciplină de aprofundare, DSI – disciplină de sinteză, DCA – disciplină de cunoaștere avansată

** DOB – disciplină obligatorie/impusă; DOP – disciplină opțională; DFA – Disciplină facultativă

3. Timpul total (ore pe semestru)

3.1 Număr de ore activități directe pe săptămână	2	din care: 3.2 curs	1	3.3 aplicații*** L	2
3.4 Total ore activități directe pe semestru	36	din care: 3.5 curs	12	3.6 aplicații L-Laborator	24
3.7 Total ore de studiu individual					139
Distribuția fondului de timp					[ore]
Studiul cărților, manualelor, suportului de curs,, notițelor, bibliografie minimală recomandată					25
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					25
Pregătire seminar / laborator / proiect, teme, referate, portofolii și eseuri					45
Pregătire pentru prezentări sau verificări					20
Pregătire pentru examinarea finală					20
Alte activități: consultații					4
3.8 Total ore pe semestru	175				
3.9 Numărul de credite	7				

*** S - seminar; L - laborator; P - proiect



4. Precondiții (acolo unde este cazul)

4.1 de curriculum	Arhitectura sistemelor de calcul, Retele de calculatoare, Structuri de date, Algoritmi fundamentali, Programare orientată pe obiect, Tehnici avansate de programare, Ingineria sistemelor soft
4.2 de rezultate ale învățării	Elaborarea și dezvoltarea unor soluții software complexe utilizând algoritmi eficienți și paradigme diverse de programare Proiectarea, planificarea, construirea, dezvoltarea unor aplicații software scalabile și utilizarea eficientă a resurselor hardware și software. Inițierea, pregătirea, realizarea, propunerea unor metode de dezvoltare a proiectelor informatice complexe; realizarea unor rapoarte profesionale specifice.

5. Condiții necesare pentru desfășurarea optimă a activităților didactice (acolo unde este cazul)

5.1. de desfășurare a cursului	Sala de curs disponibilă
5.2. de desfășurare a laboratorului	Sala de laborator disponibilă

6. Obiectivele disciplinei

6.1 Obiectivul general al disciplinei	Familiarizarea studenților cu elemente de optimizare ale calculului din aplicații ce rulează pe sisteme distribuite
6.2 Obiectivele specifice	<ul style="list-style-type: none">• Cunoașterea și utilizarea adecvată a noțiunilor specifice sistemelor paralele și distribuite• Cunoașterea și înțelegerea principiilor, etapelor și a tehnicilor de realizare a aplicațiilor paralele și distribuite• Interpretarea și soluționarea optimă a problemelor ce impun rezolvări folosind programarea paralelă și/sau distribuită• Identificarea și utilizarea corespunzătoare a tehnicilor de programare ce trebuie folosite pentru dezvoltarea algoritmilor paraleli și distribuiți• Analiza logică a problemelor științifice ce necesită calcul paralel și distribuit, formulând modele de analiză și implementare.

7. Rezultatele învățării

Cunoștințe	Studentul/absolventul <ul style="list-style-type: none">• stie să identifice tipurile și structurile modelelor distribuite• stie să argumenteze alegerea tipului de model, în funcție de interacțiunile dintre fenomenele și procesele modelate• analizează și evaluează strategiile de optimizare pentru comunicare, sincronizare și toleranță la erori pentru a îmbunătăți eficiența aplicațiilor paralele/distribuite.
------------	---



Aptitudini	<p>Studentul/absolventul</p> <ul style="list-style-type: none"> poate construi un sistem distribuit. poate valida modelele de comunicatii si poate face analiza lor critica prin comparatie cu sistemele reale de comunicatii. poate proiecta un model de comunicatii ce asigura securitatea datelor. proiectează și dezvoltă programe paralele folosind tehnici de optimizare care minimizează timpul de execuție și maximizează utilizarea resurselor. aplică și testează algoritmi de optimizare distribuită în scenarii reale, comparând și ajustând performanța acestora pentru aplicații paralele scalabile.
Responsabilitate și autonomie	<p>Studentul/absolventul</p> <ul style="list-style-type: none"> are o atitudine de discernământ și probitate în utilizarea surselor de informare. indeplinește la termen, în mod riguros, eficient și responsabil, sarcini profesionale cu grad ridicat de complexitate, în condiții de autonomie decizională, cu respectarea riguroasă a deontologiei profesionale.

8. Conținuturi

8.1 Curs	Metode de predare	Număr ore alocate
Sisteme distribuite (definitie, componente, paradigme de comunicare, structuri arhitecturale)	Medode de predare- invatare interactive;	2
Programare paralela in sisteme distribuite	Metode care implica activ studentii în învățare, punându-i în situația de a realiza conexiuni logice, de a produce idei și opinii proprii argumentate Problematizarea; Conversatia; Metodele active Sintetiza/ esențializarea informațiilor Învățarea independentă și prin cooperare	2
Modelul de programare cu transfer de mesaje.		2
Biblioteca MPI		2
Programare multi-thread in Java		2
Operații Big Data folosind paradigma de programare MapReduce		2
Cloud programming in memorie Apache SPARK		2
Algoritmi de optimizare in mediu distribuit		2



Bibliografie

1. Ernest Scheiber, Programare concurenta si paralel-distribuita in Java, Ed. Albastra, 2018
2. Ilie Popa, Petre Anghelescu Algoritmi de programare paralela si distribuita. Lucrari practice, MatrixRom
3. George Coulouris, Jean Dollimore and Tim Kindberg, Introduction to Distributed Systems-Concepts and design, Prentice Hall, ISBN 0201-619-180, 2008
4. M. Singhal, A Kshemkalyani, Distributed computing: principles, algorithms and systems, University Press, 2008
5. Note de curs, <https://sites.google.com/site/cgherghina>

8.2 Laborator

	Metode de predare	Număr ore alocate
Programare paralelă bazat pe transfer de mesaje . Biblioteca MPI.	Dialogul;	8
Programare multi-thread in Java	Problematizarea;	2
Operații Big Data folosind paradigma de programare MapReduce	Metodele active și interactive cu multiple;	4
Cloud programming in memorie Apache SPARK	Sintetiza/esențializarea informațiilor;	6
Procesare paralela in context distribuit - algoritmi de calcul – descriere, optimizare, discutia rezultatelor	Invățarea independentă și prin cooperare. Exercițiul	4

Bibliografie

1. Calcul paralel și distribuit - Lucrari practice, C. Ivan, Editura UTPress, ISBN ISBN 978-973-662-283-0, 2007
2. Tehnologii și aplicații în calcul paralel și distribuit, Editura UTPress
3. Peter Pacheco, Parallel Programming with MPI 1st Edition, Morgan Kaufmann, 1996
4. Using MPI, third edition: Portable Parallel Programming with the Message-Passing Interface (Scientific and Engineering Computation) 3rd ed., The MIT Press, 2014
5. Marko Boger, Java in Distributed Systems: Concurrency, Distribution and Persistence 1st Edition, Wiley, 2001
6. Martin Kalin, Java Web Services: Up and Running: A Quick, Practical, and Thorough Introduction Second Edition, O'Reilly Media, 2013
7. Note de laborator, <https://sites.google.com/site/cgherghina>

9. Evaluare

Tip activitate	9.1 Criterii de evaluare	9.2 Metode de evaluare	9.3 Pondere din nota finală
9.4 Curs	Cunoașterea aspectelor teoretice referitoare la programare paralela in sisteme distribuite	Examen (test grila cu variante multiple de răspuns/eseu)	50%



UOC-PO-10 Anexa 3

9.5 Laborator	Interes și capacitate de lucru pentru studiu individual și în echipă Problematizare Cunoașterea aspectelor teoretice și practice referitoare la sisteme distribuite și programarea optimă pe aceste sisteme.	Test scris– proiectarea și implementarea unor aplicații de programare distribuită/paralela	50%
Toate notele obținute pe parcursul semestrului și la evaluarea finală includ un punct din oficiu.			
9.6 Standard minim de performanță / Condiții de promovare			
Caracteristici esențiale ale serviciilor web / componentelor EJB, metrici de performanță ale algoritmilor paraleli, realizarea unei aplicații distribuite (RMI/EJB, Web Service) cu grad de dificultate redus			
Activitățile și rezultatele evaluării continue din timpul semestrului sunt finale și nu fac obiectul reevaluării în sesiunile de examen sau restante.			
Examenul restanță aplică același procent de contribuție la nota finală, așa cum este menționat mai sus.			
Toate regulile de mai sus se aplică studenților din anii precedenți.			

Data completării,

Titular activităților de curs,

Titular aplicații,

15.09.2025

Conf.univ.dr. Serban Cristina

Conf.univ.dr.Serban Cristina

Data avizării în Departament,
19.09.2025

Director de Departament,
Conf. univ. dr. Pelican Elena

Decan,
Conf.univ.dr. Nicola Aurelian