

FIȘA DISCIPLINEI ¹⁾

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Petrol-Gaze din Ploiești
1.2. Facultatea	Litere și Științe
1.3. Departamentul	Informatică, Tehnologia Informației, Matematică și Fizică
1.4. Domeniul de studii universitare	Informatică
1.5. Ciclul de studii universitare	Licență
1.6. Programul de studii universitare	Informatică

2. Date despre disciplină

2.1. Denumirea disciplinei	Curs opțional 3 – Programare Paralelă, Concurrentă și Distribuită
2.2. Titularul activităților de curs	Conf.dr.ing. Zoran Constantinescu
2.3. Titularul activităților seminar/laborator	inf. Adrian Rădulescu
2.4. Titularul activității proiect	-
2.5. Anul de studiu	III
2.6. Semestrul *	5
2.7. Tipul de evaluare	C
2.8. Categoria formativă** / regimul*** disciplinei	DS/A

* numărul semestrului este conform planului de învățământ;

** DF - Discipline fundamentale; DD - discipline de domeniu; DS - discipline de specialitate; DC - discipline complementare, DA - disciplina de aprofundare, DSI- disciplina de sinteza. *** obligatorie = O; opțională = A; facultativă = L

3. Timpul total estimat (ore pe semestru al activităților didactice)

3.1. Număr de ore pe săptămână	4	din care: 3.2. curs	2	3.3. Seminar/laborator	2	3.4. Proiect	-
3.5. Total ore din planul de învățământ	56	din care: 3.6. curs	28	3.7. Seminar/laborator	28	3.8. Proiect	-
3.9. Distribuția fondului de timp							ore
Studiu după manual, suport de curs, bibliografie și notițe							20
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren							20
Pregătire seminar/laboratoare, teme, referate, portofolii și eseuri							29
Tutoriat							-
Examinări							-
Alte activități – elaborare proiect							-
3.10 Total ore studiu individual	69						
3.11. Total ore pe semestru	125						
3.12. Numărul de credite	5						

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	➤ Arhitectura Sistemelor de Calcul, Fundamentele Programării, Sisteme de Operare, Rețele de Calculatoare, Elemente Avansate de Sisteme de Operare și Rețele de Calculatoare
4.2. de competențe	➤ Cunoașterea arhitecturii sistemelor de calcul, sistemelor de operare și rețelelor de calculatoare, precum și a fundamentelor programării.

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	➤ sală de curs multimedia cu videoproiector și conexiune la Internet
5.2. de desfășurare seminar/laborator	➤ sală de laborator care să permită desfășurarea de sesiuni de lucru Linux

¹⁾ Adaptare după Ordinul Ministrului educației, cercetării, tineretului și sportului nr. 5 703/2011 privind implementarea Codului național al calificărilor din învățământul superior, publicat în Monitorul Oficial al României, partea I, nr.880 bis / 13.XII.2011

6. Competențe specifice acumulate

Competențe profesionale	<ul style="list-style-type: none"> ➤ C3.1 Descrierea de concepte, teorii și modele folosite în domeniul de aplicare; ➤ C3.2 Identificarea și explicarea modelelor informatice de bază adecvate domeniului de aplicare. ➤ C3.3 Utilizarea modelelor și instrumentelor informatice și matematice pentru rezolvarea problemelor specifice domeniului de aplicare. ➤ C3.4 Analiza datelor și a modelelor. ➤ C3.5 Elaborarea componentelor informatice ale unor proiecte interdisciplinare. ➤ Descrierea conceptuală detaliată a structurii și funcționalității diverselor tipuri de aplicații și infrastructuri pentru procesarea specifică a informației ➤ Utilizarea de teorii, modele conceptuale, instrumente și infrastructuri specifice pentru explicarea structurii și funcționalității unei largi varietăți de aplicații informatice ➤ Identificarea modelelor, metodelor și instrumentelor adecvate pentru rezolvarea unor probleme reale (aplicații, infrastructuri); ➤ Alegerea criteriilor, metodelor și instrumentelor de evaluare a calității performanțelor și limitărilor în procesarea specifică a informațiilor (aplicații, infrastructuri); ➤ Dezvoltarea de proiecte informatice specifice.
Competențe transversale	<ul style="list-style-type: none"> ➤ CT1. Aplicarea regulilor de muncă organizată și eficientă, a unor atitudini responsabile față de domeniul didactic, științific și profesional, în vederea valorificării creative a propriului potențial, cu respectarea principiilor și normelor de etică profesională; ➤ CT2. Desfășurarea eficientă a activităților organizate în echipă și dezvoltarea capacităților empatică și de comunicare inter-personală, de relaționare și colaborare cu persoane și grupuri diverse implicate în dezvoltarea și utilizarea de sisteme software; ➤ CT3. Utilizarea de metode și tehnici eficiente de învățare, informare, cercetare și dezvoltare a capacităților de valorificare a cunoștințelor, dar și de adaptare la cerințele unei societăți dinamice și în continuă schimbare, precum și dezvoltarea capacității de a comunica eficient și profesionist atât în limba română, cât și într-o limbă de circulație internațională, prin însușirea și folosirea adecvată a limbajului de specialitate.

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

7.1. Obiectivul general al disciplinei	<ul style="list-style-type: none"> ➤ <i>Dobândirea de către studenți a cunoștințelor fundamentale teoretice și practice privind sistemele software de tip paralel, concurent și distribuit, precum și a modului de proiectare și dezvoltare a acestora. Astfel, cursul oferă noțiuni fundamentale despre: bazele programării paralele, concurente și distribuite; paradigme de comunicare; arhitecturi orientate pe aplicații; paradigme de programare paralelă; Message Passing Interface (MPI); sisteme paralele de date; algoritmi paraleli; parallel Python; Open MP; evaluarea și optimizarea performanțelor. Aplicații specifice</i>
7.2. Obiectivele specifice	<p><i>După parcurgerea disciplinei studenții vor putea să:</i></p> <ul style="list-style-type: none"> ➤ Identifice și să descrie principalele componente ale unui sistem software de tip paralel, concurent, respectiv distribuit; ➤ Înțeleagă, să evalueze critic și să folosească paradigme de programare de tip paralel (de exemplu, threads, OpenMP, MPI); ➤ Înțeleagă și să abordeze adecvat problemele de sincronizare și comunicație, precum și evaluarea și optimizarea performanțelor în sistemele paralele.

8. Conținuturi

8.1. Curs	Nr. ore	Metode de predare	Obs.
1. Introducere. Noțiuni generale despre sisteme paralele, concurente, distribuite.	2	<ul style="list-style-type: none"> • prelegeri active și angajante; • învățare centrată pe student; • învățare hibridă, cu folosirea resurselor educaționale open; • învățare reflectivă etc. 	
2. Tehnologii care facilitează dezvoltarea sistemelor paralele. Sisteme multiprocesor și procesoare multicore.	4		
3. Programare concurentă multi-threading.	4		
4. Paradigme de programare paralelă, concurentă și distribuită (multithreading, paralelismul datelor, comunicație prin mesaje).	4		
5. Algoritmi de comunicație.	4		
6. Algoritmi paraleli. Performanțele algoritmilor paraleli. Tehnici de paralelizare.	6		
7. Aplicații ale calculului paralel, concurent și distribuit. Evaluarea și optimizarea performanțelor.	4		
Bibliografie			
1. Foster, I.: Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering, Pearson, 2019* 2. Garg, V.: Elements of Distributed Computing, Wiley, 2015.* 3. Grama, A.: Introduction to parallel computing, Pearson Addison Wesley, 2003* 4. Kirk, D., Hwu, W.-M.: Programming Massively Parallel Processors. A Hands-On Approach, Elsevier, 2010.** 5. Quinn, M.: Parallel Programming in C with MPI and OpenMP, McGraw Hill Education, 2004. ** 6. Sanders, G., Kandrot, E.: CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley, 2010* 7. Zaccone, G.: Python Parallel Programming Cookbook: Over 70 recipes to solve challenges in multithreading and distributed system with Python 3, 2nd edition, Packt Publishing, 2019* 8. Resurse educaționale disponibile la http://www.unde.ro/cursuri/PPCD/ și http://www.unde.ro/cursuri/OCW/ * Disponibile la biblioteca departamentului iTIMF ** Disponibile la CERTIMF (L4)			
8.2. Seminar / laborator	Nr. ore	Metode de predare	Obs.
1. Programare IPC și multithreading în limbajul C. Metode de sincronizare între thread-uri.	8	<ul style="list-style-type: none"> • prezentări active și angajante; învățarea prin descoperire; învățare bazată pe rezolvarea de probleme; învățare centrată pe student; learning by doing; învățare hibridă, cu folosirea resurselor educaționale deschise; învățare reflectivă etc. 	
2. Programare paralelă în OpenMP.	8		
3. Programare paralelă folosind comunicația prin mesaje (MPI).	8		
4. Evaluarea performanțelor programelor paralele.	2		
5. Aplicații din lumea reală.	2		
Bibliografie: idem curs			
8.3. Proiect	Nr. ore	Metode de predare	Observații
-			

9. Coroborarea conținuturilor disciplinei cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatori reprezentativi din domeniul aferent programului

- Conținuturile disciplinei corespund cu așteptările reprezentanților comunității epistemice, asociațiilor profesionale și angajatorilor reprezentativi din domeniul aferent programului, așa după cum rezultă din prezenta fișă, dar și din fișa specializării, acestea fiind în concordanță deplină cu CNCIS și COR;
- Disciplina de față respectă recomandările IEEE/CS și ACM legate de planul de învățământ și de conținuturile necesare pentru specializarea Informatică/Știința Calculatoarelor;
- Discipline având conținuturi educaționale asemănătoare există în planurile de învățământ pentru specializarea Informatică ale marilor universități din România și din străinătate.

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs	Dobândirea competențelor profesionale și transversale specifice disciplinei	<i>Evaluare orală, pe baza temelor de casa</i>	40%
10.5. Seminar /laborator		<i>Evaluare prin proiect sau referat.</i>	50%
10.6. Proiect			Din oficiu 10%
10.7. Standard minim de performanță			
<ul style="list-style-type: none"> ➤ <i>Identificarea și descrierea adecvată a principalelor componente pentru sistem de tip paralel/concurent/distribuit;</i> ➤ <i>Înțelegerea și descrierea adecvată a paradigmatelor de programare de tip paralel/concurent/distribuit;</i> ➤ <i>Realizarea unui proiect informatic specific de complexitate redusă.</i> 			

Data completării

Semnătura titularului de curs

Semnătura titularului de seminar/laborator

Semnătura titularului de proiect

22.09.24

Data avizării în departament

Director de departament
(funcție didactică, nume, prenume)
(Semnătură)

Decan
(funcție didactică, nume, prenume)
(Semnătură)

24.09.24