Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti

Facultatea de Electronică, Telecomunicaţii şi

Tehnologia Informaţiei

FIŞA DISCIPLINEI

1. Date despre program

1.1 Instituţia de învăţământ superior	**Universitatea Naţională de Ştiinţă şi Tehnologie Politehnica Bucureşti**
1.2 Facultatea	**Electronică, Telecomunicaţii şi Tehnologia Informaţiei**
1.3 Departamentul	**Electronică Aplicată şi Ingineria Informaţiei**
1.4 Domeniul de studii	Calculatoare şi Tehnologia Informaţiei
1.5 Ciclul de studii	Licenţă
1.6 Specializarea	Ingineria Informaţiei

2. Date despre disciplină

2.1 Denumirea disciplinei (ro) (en)				Algoritmi paraleli şi distribuiţi 1 Parallel and distributed algorithms 1			
2.2 Titularul activităţilor de curs				S.l./Lect. Dr. George Valentin STOICA			
2.3 Titularul activităţilor de seminar / laborator				S.l./Lect. Dr. George Valentin STOICA, S.l./Lect. Dr. Elena Cristina STOICA			
2.4 Anul de studiu	4	2.5 Semestrul	I	2.6. Tipul de evaluare	E	2.7 Regimul disciplinei	Ob
2.8 Tipul disciplinei		D	2.9 Codul disciplinei	04.D.07.O.102		2.10 Tipul de notare	Nota

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

3.1 Număr de ore pe săptămână	3	Din care: 3.2 curs	2.00	3.3 seminar/laborator	1
3.4 Total ore din planul de învăţământ	42.00	Din care: 3.5 curs	28	3.6 seminar/laborator	14
Distribuţia fondului de timp:					ore
Studiul după manual, suport de curs, bibliografie şi notiţe Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate Pregătire seminarii/ laboratoare/proiecte, teme, referate, portofolii şi eseuri					30
Tutorat					0
Examinări					3
Alte activităţi (dacă există):					0

3.7 Total ore studiu individual	33.00
3.8 Total ore pe semestru	75
3.9 Numărul de credite	3

4. Precondiţii (acolo unde este cazul)

4.1 de curriculum	Parcurgerea urmatoarelor discipline: - Programarea calculatoarelor (PC) - Structuri de date si algoritmi (SDA) - Programare obiect-orientata (POO) - Arhitectura sistemelor de calcul (ASC) - Sisteme de operare (SO)

4.2 de rezultate ale învățării	Cunoştinţe generale de manipulare a sistemului de operare Linux. Cunoştinţe generale de programare (cunoaşterea limbajelor C, C++), algoritmi, arhitectura sistemelor de calcul, sisteme de operare.

5. Condiţii necesare pentru desfăşurarea optimă a activităţilor didactice (acolo unde este cazul)

5.1 Curs	Nu este cazul.
5.2 Seminar/ Laborator/Proiect	Prezența la laborator (conform regulamentului studiilor universitare în UPB).

6. Obiectiv general *(Se referă la intenţiile profesorilor pentru studenţi, la ceea ce studenţii vor fi învăţaţi în timpul cursului. Oferă o orientare cu privire la locul cursului în cadrul domeniului ştiinţific abordat, precum şi la rolul pe care acesta îl are în cadrul specializării studiate. Vor fi descrise de o manieră generală tematicile abordate, justificarea includerii cursului în planul de învăţământ al specializării studiate etc.)*

Cursul familiarizează studenţii cu problemele calculului paralel si ale principiilor de dezvoltare a algoritmilor paraleli pentru rezolvarea problemelor intens computationale. Sunt studiate modalităţile de analiză, proiectare, implementare şi testare a algoritmilor paraleli în sisteme multiprocesor şi în clustere de calculatoare folosind limbaje şi biblioteci specifice.
Aplicatiile practice de laborator au ca obiectiv cunoaşterea de catre studenti a modului de lucru în sisteme paralele de calcul. Studentii învaţă să proiecteze şi să implementeze algoritmi paraleli folosind limbaje şi biblioteci specifice (C, C++, limbajul R, bibliotecile Pthread, OpenMP, MPI) în sisteme cu memorie partajată şi cu transfer de mesaje.

7. Competenţe *(Capacitatea dovedită de a utiliza cunoştinţe, aptitudini şi abilităţi personale, sociale şi/sau metodologice în situaţii de muncă sau de studiu şi pentru dezvoltarea profesională şi personală. Reflectă cerinţele angajatorilor.)*

Specifice	Aplicarea cunoştinţelor, conceptelor şi metodelor elementare privitoare la arhitectura sistemelor de calcul, microcontrolere, limbaje şi tehnici de programare. Dezvoltarea sistemelor software complexe: sisteme de baze de date, sisteme paralele şi distribuite, sisteme multimedia, interfeţe om- maşină. Proiectarea şi utilizarea sistemelor de calcul şi a reţelelor de calculatoare
Transversale (generale)	Capacitatea de a asigura planificarea şi managementul proiectelor din domeniul ingineriei informaţiei. Capacitatea de a lua decizii în vederea rezolvării problemelor curente, sau imprevizibile, care apar în procesul de exploatare a sistemelor de calcul. Capacitatea de a se informa şi documenta permanent pentru dezvoltarea personală şi profesională prin citirea literaturii de specialitate. Capacitatea de a comunica şi de a prezenta conţinut tehnic atât în limba română, cât şi în limba engleză. Flexibilitate în utilizarea de noi sisteme şi tehnologii în cadrul unei echipe în care membrii împreună ating un obiectiv bine definit, asumând în acelaşi timp roluri sau sarcini diferite. Capacitatea de a lua decizii în vederea rezolvării problemelor curente, sau imprevizibile, care apar în procesul de exploatare a sistemelor de calcul.

8. Rezultatele învăţării *(Sunt enunțuri sintetice referitoare la ceea ce un student va fi capabil să facă sau să demonstreze la finalizarea unui curs. Rezultatele învățării reflectă realizările studentului și mai puțin intențiile profesorului. Rezultatele învățării informează studenții despre ceea ce se așteaptă de la ei din punct de vedere al performanței, pentru a obține notele și creditele dorite. Sunt definite în termeni concreți, folosind verbe similare exemplelor de mai jos și indică ceea ce se va urmări prin evaluare. Rezultatele învățării vor fi astfel redactate încât să fie evidențiată clar relația față de competențele definite la punctul 7.)*

Cunoştinţe	*Rezultatul asimilării de informații prin învățare. Cunoștințele reprezintă ansamblul de fapte, principii, teorii și practici legate de un anumit domeniu de muncă sau de studiu. Pot fi teoretice și/sau faptice.* **Enumeră** cele mai importante etape care au marcat dezvoltarea domeniului calculului paralel. **Defineşte** noțiuni specifice domeniului de calcul paralel precum: accelerarea paralela, eficienta paralela, costul paralel, costul suplimentar, planificari statice, planificari dinamice ale proceselor/thread-urilor etc. **Descrie** noțiuni de bază de proiectare ale sistemelor paralele si distribuite. **Implementează algoritmii studiaţi cu ajutorul bibliotecilor POSIX, OpenMP(pentru sistemele paralele) şi MPI(pentru sistemele distribuite).** **Cunoştinţele acumulate în cadrul orelor de curs si laborator pot fi folosite în proiectarea si implementarea** algoritmilor paraleli şi distribuiți.
Aptitudini	*Capacitatea de a aplica cunoștințe și de a utiliza know-how pentru a duce la îndeplinire sarcini și a rezolva probleme. Aptitudinile sunt descrise ca fiind cognitive (implicând utilizarea gândirii logice, intuitive și creative) sau practice (implicând dexteritate manuală și utilizarea de metode, materiale, unelte și instrumente).* **Selectează** şi grupează informații relevante din domeniul calculului paralel. **Utilizează argumentat principii specifice în vederea** proiectării şi implementării algoritmilor paraleli şi distribuiți in sisteme paralele şi distribuite. **Lucrează** productiv în echipă. **Verifică experimental soluţii identificate.** **Rezolvă** aplicații practice. **Analizează şi compară** mai mulți algoritmi paraleli şi distribuiți din punctul de vedere al parametrilor principali de performanță (accelerare, eficiență, cost, izoeficiență etc.). **Identifică soluţii** şi elaborează planuri de proiectare a algoritmilor paraleli şi distribuiți pentru bibliotecile POSIX, OpenMP şi MPI. **Formulează** concluzii la implementările realizate. **Argumentează** soluțiile identificate precum şi modurile de rezolvare a algoritmilor propuşi.

Responsabilitate și autonomie	*Capacitatea cursantului de a aplica în mod autonom și responsabil cunoștințele și aptitudinile sale.* **Selectează** surse bibliografice potrivite şi le analizează. **Respectă principiile de etică academică,** citând corect sursele bibliografice utilizate. **Demonstrează receptivitate** pentru contexte noi de învățare. **Manifestă colaborare** cu ceilalți colegi și cadre didactice în desfășurarea activităților didactice **Demonstrează autonomie** în organizarea situației/contextului de învățare sau a situației problemă de rezolvat **Manifestă responsabilitate socială** prin implicarea activă în viața socială studențească/implicare în evenimentele din comunitatea academică **Promovează/contribuie prin soluții noi, aferente domeniului de specialitate** pentru a îmbunătăți calitatea vieții sociale. **Conștientizează valoarea contribuției sale în domeniul ingineriei** la identificarea de soluții viabile/sustenabile care să rezolve probleme din viața socială și economică (responsabilitate socială). **Aplică principii de etică/deontologie profesională în analiza impactului tehnologic al soluțiilor propuse** în domeniul de specialitate asupra mediului înconjurător. **Analizează și valorifică oportunități de afaceri**/de dezvoltare antreprenorială în domeniul de specialitate. **Demonstrează** abilități de management al situațiilor din viața reală (gestionarea timpului colaborare vs. conflict).

9. Metode de predare *(Se vor avea în vedere metode care să asigure predarea centrată pe student. Se va descrie modul în care se asigură participarea studenților la stabilirea propriului parcurs de învățare, cum se identifică eventualele rămâneri în urmă și ce măsuri remediale se adoptă în astfel de cazuri.)*

Pornindu-se de analiza caracteristicilor de învățare ale studenților şi de la nevoile lor specifice, procesul de predare va explora metode de predare atât expozitive (prelegerea, expunerea), cât şi conversative-interactive, bazate pe modele de învățare prin descoperire facilitate de explorarea directa şi indirectă a realității (experimentul, demonstrația, modelarea), dar şi pe metode bazate pe acțiune, precum exercițiul, activitățile practice şi rezolvarea de probleme.
În activitatea de predare vor fi utilizate prelegeri, în baza unor prezentări Power Point sau diferite filmulețe care vor fi puse la dispoziția studenților. Fiecare curs va debuta cu recapitularea capitolelor deja parcurse, cu accent asupra noțiunilor parcurse la ultimul curs.
Prezentările utilizează imagini şi scheme, astfel încât informațiile prezentate să fie uşor de înțeles şi asimilat.
Acestă disciplină acoperă informații şi activități practice menite să-i sprijine pe studenți în eforturile de învățare şi de dezvoltare a unor relații optime de colaborare şi comunicare într-un climat favorabil învățării prin descoperire.
Se va avea în vedere exersarea abilităţilor de ascultare activă şi de comunicare asertivă, precum şi a mecanismelor de construcţie a feedback-ului, ca modalităţi de reglare comportamentală în situații diverse şi de adaptare a demersului pedagogic la nevoile de învățare ale studenților.
Se va exersa abilitatea de lucru în echipă pentru rezolvarea diferitelor sarcini de învățare.
Predarea se bazează pe folosirea videoproiectorului (acoperind funcţia de comunicare şi demonstrativă).
Cursurile sunt disponibile online pe platforma Moodle.

10. Conținuturi

CURS		
Capitolul	**Conținutul**	**Nr. ore**

1	Introducere in calculul paralel 1.1. Motivarea paralelismului: aplicatii cu cerinte de calcul ridicate 1.2. Introducerea paralelismului in arhitecturile de calcul	2
2	2. Arhitecturi de calcul paralel 2.1. Clasificarea arhitecturilor dupa fluxurile de control si de date 2.2. Clasificarea arhitecturilor după organizarea memoriei 2.3. Retele de interconectare in sistemele paralele si distribuite (retele statice si dinamice)	2
3	3. Proiectarea algoritmilor paraleli 3.1. Etapele de proiectare a algoritmilor paraleli: partitionarea calculelor si a datelor, comunicarea intre task-uri, aglomerarea (gruparea) task-urilor, alocarea task-urilor 3.2. Analiza dependentelor intre task-uri 3.3. Paralelizarea buclelor de program 3.4. Paralelizarea automată a algoritmilor 3.5. Exemple de proiectare a algoritmilor paraleli	2
4	4.Analiza si estimarea performantelor algoritmilor paraleli 4.1. Analiza si estimarea performantelor algoritmilor folosind complexitatea asimptotica 4.2. Accelerarea paralela – legea lui Amdahl 4.3. Algoritmi cost-optimali 4.5. Exemple de analiza a performantelor algoritmilor paraleli	6
5	5. Programarea algoritmilor paraleli 5.1. Clasificarea tehnicilor de programare paralelă 5.2. Crearea dinamica a proceselor si a thread-urilor 5.3. Limbaje si biblioteci de programare paralela	4
6	6. Programarea paralelă prin memorie partajată 6.1. Mecanisme de sincronizare intre procese si thread-uri: mutex-uri, semafoare, bariere de sincronizare 6.2. Biblioteca de thread-uri POSIX (PTHREAD) 6.3. Tehnologia OpenMP 6.4. Exemple de implementare a algoritmilor paraleli în sisteme cu memorie partajata	8
7	7. Programarea paralelă prin transfer de masaje 7.1. Primitive de transfer de mesaje 7.2. Biblioteci de programare prin transfer de mesaje: MPI, PVM 7.3. Exemple de implementare a algoritmilor paraleli in sisteme cu transfer de mesaje	4
	Total:	28

Bibliografie:
[1]. PRINCIPIILE CALCULULUI PARALEL, Felicia Ionescu, EDITURA TEHNICĂ, Bucureşti, 1999

LABORATOR

Nr. crt.	Conţinutul	Nr. ore
1	Laborator 1. IMPLEMENTAREA ALGORITMILOR PARALELI. CONDIŢII DE PARALELISM.	3

2	Laborator 2. IMPLEMENTAREA ALGORITMILOR PARALELI. ACCESUL CONCURENT LA RESURSE	3
3	Laborator 3. IMPLEMENTAREA ALGORITMILOR PARALELI. PARALELISM LA NIVEL DE DATE. Grade de paralelism. Granularitatea de calcul	3
4	Laborator 4. IMPLEMENTAREA ALGORITMILOR PARALELI. Reducerea paralela. Sortarea. RECURSIVITATE	3
5	Laborator 5. IMPLEMENTAREA ALGORITMILOR PARALELI. Sortarea. RECURSIVITATE	3
6	Laborator 6. Algoritmi paraleli complexi	3
7	Laborator 7. Colocviu.	3
	Total:	21

Bibliografie:

[1]. PRINCIPIILE CALCULULUI PARALEL, Felicia Ionescu, EDITURA TEHNICĂ, Bucureşti, 1999
[2]. C# Programming Guide – Thread Class https://docs.microsoft.com/en-us/dotnet/api/system.threading.thread?view=netframework-4.8

11. Evaluare

Tip activitate	11.1 Criterii de evaluare	11.2 Metode de evaluare	11.3 Pondere din nota finală
11.4 Curs	- Cunoaşterea noţiunilor teoretice fundamentale - Aplicarea noţiunilor teoretice la practica proiectării algoritmilor paraleli - Analiza algoritmilor paraleli şi interpretarea rezultatelor	Examen oral în cursul sesiunii, susţinut la data fixată de comun acord; subiectele acoperă întreaga materie, realizând o sinteză între partea teoretică şi partea practică de proiectare a algoritmilor paraleli	50%
11.5 Seminar/laborator/proiect	- Cunoaşterea modului de proiectare şi implementare a algoritmilor paraleli în multiprocesoare şi clustere de calculatoare; - Cunoaşterea modului de analiză a algoritmilor paraleli şi interpretarea rezultatelor obţinute	Colocviu final de laborator constând în verificarea modului de rezolvare (proiectare, implementare, testare, analiză) de către student a unei probleme practice.	50%
11.6 Condiţii de promovare			
Obţinerea a 50% din punctajul total. Obţinerea a 50% din punctajul aferent activităţii de laborator pe parcursul semestrului.			

12. Coroborarea conţinutului disciplinei cu aşteptările reprezentanţilor angajatorilor şi asociaţiilor profesionale reprezentative din domeniul aferent programului, precum şi cu stadiul actual al cunoaşterii în domeniul ştiinţific abordat şi practicile în instituţii de învăţământ superior din Spaţiul European al Învăţământului Superior (SEÎS)

Prelucrarea informației necesită cunoașterea arhitecturilor disponibile în prezent pe piață, a limbajelor de programare, a structurilor de date necesare precum și a bibliotecilor existente adaptate diverselor domenii precum procesarea imaginilor, viziunea computerizată, inteligența artificială, grafica 3D, calculele matematice, algebra. Înțelegerea acestor aspecte împreună cu particularitățile fiecărei arhitecturi hardware creează premisele pentru implementări eficiente, optimizate și adaptate la cerințele de calcul din ce în ce mai mari odată cu complexitatea tot mai mare a aplicațiilor și a datelor acestora. Procesarea rapidă a informațiilor necesită utilizarea unor platforme adaptate, indiferent dacă sunt folosite arhitecturi cloud, arhitecturi GPU sau platforme mobile.

Data completării	Titular de curs	Titular(i) de aplicații
21.09.2026	S.l./Lect. Dr. George Valentin STOICA	S.l./Lect. Dr. George Valentin STOICA
		S.l./Lect. Dr. Elena Cristina STOICA

Data avizării în departament	Director de departament
	Conf.Dr.Ing. Bogdan Cristian FLOREA

Data aprobării în Consiliul Facultății	Decan
22.09.2026	Prof. Dr. Mihnea Udrea