

## FIȘA DISCIPLINEI

Programare Paralelă si Distribuită

Anul universitar 2026/2027

### 1. Date despre program

1.1. Instituția de învățământ superior	Universitatea Babeș Bolyai
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Departamentul de Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii / Calificarea	Informatică
1.7. Forma de învățământ	Cu frecvență

### 2. Date despre disciplină

2.1. Denumirea disciplinei	<b>Programare Paralelă si Distribuită</b>			Codul disciplinei	<b>MLR5077</b>
2.2. Titularul activităților de curs	Lect. dr. Radu Lupșa				
2.3. Titularul activităților de seminar	Lect. dr. Radu Lupșa				
2.4. Anul de studiu	3	2.5. Semestrul	5	2.6. Tipul de evaluare	Examen
2.7. Regimul disciplinei	Obligativu		2.8. Tipul disciplinei	Disciplină fundamentală (DF)	

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

3.1. Număr de ore pe săptămână	5	din care: 3.2. curs	2	3.3. seminar/ laborator/ proiect	0/2 /1
3.4. Total ore din planul de învățământ	70	din care: 3.5. curs	28	3.6 seminar/laborator	42
<b>Distribuția fondului de timp pentru studiul individual (SI) și activități de autoinstruire (AI)</b>					<b>ore</b>
Studiul după manual, suport de curs, bibliografie și notițe (AI)					10
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					10
Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					20
Tutoriat (consiliere profesională)					10
Examinări					5
Alte activități					-
<b>3.7. Total ore studiu individual (SI) și activități de autoinstruire (AI)</b>				<b>55</b>	
<b>3.8. Total ore pe semestru</b>				<b>125</b>	
<b>3.9. Numărul de credite</b>				<b>5</b>	

### 4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Fundamentele programării, Programare orientată obiect, Structuri de date și algoritmi, Sisteme de operare
4.2. de competențe	Abilități de programare

### 5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs cu videoproiector
5.2. de desfășurare a seminarului/ laboratorului	Sală cu videoproiector; stații cu medii de dezvoltare pentru C++, Python, Java și C#

### 6.1. Competențele dobândite în urma absolvirii programului de studii (se preiau din planul de învățământ)<sup>1</sup>

Competențe profesionale	
Codul competenței	Competență
CP5	definește cerințe tehnice
CP11	utilizează șabloane de proiectare de software
Competențe transversale	
Codul competenței	Competență
CT2	Soluționează probleme
CT3	Gândește analitic

### 6.2. Rezultatele învățării specifice programului de studii (se preiau din planul de învățământ)<sup>2</sup>

Rezultatele învățării vizate prin disciplină		
Codul competenței	Cunoștințe și înțelegere (Knowledge and understanding)	Abilități academice specifice (Specific academic skills)
CP1 CP6	1. Studentul/absolventul identifică, explică și argumentează concepte fundamentale de structuri de date, algoritmi și paradigme de programare, precum și a arhitecturii calculatoarelor.	1. Studentul/absolventul elaborează, dezvoltă și demonstrează soluții software complexe utilizând algoritmi eficienți și paradigme diverse de programare.
CP9 CP10	2. Studentul/absolventul alege, descrie, analizează și explică paradigmele moderne de programare, inclusiv programarea funcțională, orientată pe obiect și paralelă, utilizând limbaje și framework-uri actuale.	2. Studentul/absolventul proiectează, planifică, construiește, dezvoltă aplicații software scalabile și utilizează eficient resursele hardware și software.

### 7. Rezultatele învățării specifice disciplinei

Cunoștințe și înțelegere (Knowledge and understanding)
1. Studentul/absolventul înțelege problemele ridicate de aplicațiile paralele și distribuite
2. Studentul/absolventul cunoaște mecanismele de baza și framework-urile utilizate în construcția aplicațiilor paralele și distribuite
3. Studentul/absolventul cunoaște pattern-urile utilizate în dezvoltarea aplicațiilor paralele și distribuite
Abilități academice specifice (Specific academic skills)
1. Studentul/absolventul este capabil să proiecteze algoritmi și aplicații distribuite
2. Studentul/absolventul înțelege problemele ridicate de paralelizare și e capabil să găsească soluții la acestea
3. Studentul/absolventul e capabil să determine îmbunătățirile de performanță obținute prin paralelizare.

### 8. Conținuturi

<sup>1</sup> Se vor prelua din Planul de învățământ al programului de studii acele competențe profesionale și/sau transversale la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa disciplinei. Pentru fiecare competență se va prelua întregul enunț, inclusiv codul competenței, cu formularea care apare în planul de învățământ, fără modificări. Dacă nu se preia nici o competență din oricare din cele două categorii, se șterge linia din tabel aferentă acelei categorii.

<sup>2</sup> Se menționează rezultatele învățării specifice programului de studiu la dezvoltarea cărora contribuie disciplina pentru care se elaborează fișa. Enunțurile, preluate fără modificări din Planul de învățământ în funcție de tipul disciplinei (DF/DS/DC) se trec în dreptul competenței asociate.

<b>8.1 Curs</b>	<b>Metode de predare - învățare</b>	<b>Observații<sup>3</sup></b>
C1. Introducere. Necesitatea paralelismului. Programare concurenta vs paralela vs distribuită.	Expunere: descriere, explicații, exemple, dezbateri.	
C2. Arhitecturi paralele: pipeline, mașini vectoriale, computere grid și cluster.	Expunere: descriere, explicații, exemple, dezbateri.	
C3. <i>Thread</i> -uri. <i>Race conditions</i> , exclusiune mutuală, interblocare ( <i>deadlock</i> ). Primitive de sincronizare.	Expunere: descriere, explicații, exemple, dezbateri.	
C4. Paralelism producător-consumator. Primitive de nivel coborât (variabile condiționale) și de nivel înalt ( <i>futures</i> , cozi producător-consumator).	Expunere: descriere, explicații, exemple, dezbateri.	
C5-C6. Programare asincronă. <i>Futures</i> cu continuări. Corutine.	Expunere: descriere, explicații, exemple, dezbateri.	
C7. Algoritmi paraleli simpli.	Expunere: descriere, explicații, exemple, dezbateri.	
C8. Algoritmi de descompunere recursivă și de explorare paralelă.	Expunere: descriere, explicații, exemple, dezbateri.	
C9. Programare distribuită folosind MPI.	Expunere: descriere, explicații, exemple, dezbateri.	
C10. Descompunere recursivă și explorare paralelă în varianta distribuită.	Expunere: descriere, explicații, exemple, dezbateri.	
C11. Protocoale distribuite. Ceasuri Lamport.	Expunere: descriere, explicații, exemple, dezbateri.	
C12. Memorie partajată distribuită.	Expunere: descriere, explicații, exemple, dezbateri.	
C13. Programarea pe GPGPU. OpenCL.	Expunere: descriere, explicații, exemple, dezbateri.	
C14. Toleranța la pene.	Expunere: descriere, explicații, exemple, dezbateri.	
Bibliografie <ol style="list-style-type: none"> <li>1. <a href="http://www.cs.ubbcluj.ro/~rlupsa/edu/pdp/">http://www.cs.ubbcluj.ro/~rlupsa/edu/pdp/</a></li> <li>2. Ian Foster. Designing and Building Parallel Programs, Addison-Wesley 1995.</li> <li>3. Michael McCool, Arch Robinson, James Reinders, Structured Parallel Programming: Patterns for Efficient Computation," Morgan Kaufmann,, 2012 .</li> <li>4. Berna L. Massingill, Timothy G. Mattson, and Beverly A. Sanders, Addison A Pattern Language for Parallel Programming. Wesley Software Patterns Series, 2004.</li> <li>5. Grama, A. Gupta, G. Karypis, V. Kumar. Introduction to Parallel Computing, Addison Wesley, 2003.</li> <li>6. D. Grigoras. Calculul Paralel. De la sisteme la programarea aplicatiilor. Computer Libris Agora, 2000.</li> <li>7. V. Niclescu. Calcul Paralel. Proiectare si dezvoltare formala a programelor paralele. Presa Univ. Clujana, 2006.</li> <li>8. D.B. Skillicorn, D. Talia. Models and Languages for Parallel Computation. ACM Computer Surveys, 30(2) pg.123-136, June 1998.</li> <li>9. B. Wilkinson, M. Allen, Parallel Programming Techniques and Applications Using Networked Workstations and Parallel Computers, Prentice Hall, 2002</li> <li>10. E.F. Van de Velde. Concurrent Scientific Computing. Spring-Verlag, New-York Inc. 1994.</li> <li>11. Boian F.M. Ferdean C.M., Boian R.F., Dragos R.C. Programare concurenta pe platforme Unix, Windows, Java. Ed. Albastra, grupul Microinformatica, Cluj, 2002 .</li> <li>12. OpenMP Tutorials</li> <li>13. MPI Tutorials</li> <li>14. OpenCL Tutorials</li> </ol>		
<b>8.2 Seminar / laborator</b>	<b>Metode de predare - învățare</b>	<b>Observații</b>
L1. Introducere	Dialog, dezbateri, exemple, descoperire dirijată.	

<sup>3</sup> De exemplu aspecte organizatorice, recomandări pentru studenți, aspecte specifice legate de curs/seminar cum ar fi invitarea unor practicieni în domeniu etc.

L2-L3. Primitive de sincronizare.	Dialog, dezbateri, exemple, descoperire dirijată.	
L4. Paralelism producător-consumator.	Dialog, dezbateri, exemple, descoperire dirijată.	
L5-L6. Programare asincronă.	Dialog, dezbateri, exemple, descoperire dirijată.	
L7. Algoritmi paraleli simpli.	Dialog, dezbateri, exemple, descoperire dirijată.	
L8. Descompunere recursivă.	Dialog, dezbateri, exemple, descoperire dirijată.	
L9. Explorare paralelă.	Dialog, dezbateri, exemple, descoperire dirijată.	
L10. Programare distribuită folosind MPI.	Dialog, dezbateri, exemple, descoperire dirijată.	
L11 Descompunere recursivă și explorare paralelă utilizând MPI	Dialog, dezbateri, exemple, descoperire dirijată.	
L12. Memorie partajată distribuită	Dialog, dezbateri, exemple, descoperire dirijată.	
L13. OpenCL	Dialog, dezbateri, exemple, descoperire dirijată.	
L14. Finalizarea activităților de laborator.	Dialog, dezbateri, exemple, descoperire dirijată.	
<b>Bibliografie</b> <ol style="list-style-type: none"> <li>1. Eckel, B., Thinking in Java, 4th Edition, New York: Prentice Hall, 2006.</li> <li>2. Larman, C.: Applying UML and Design Patterns: An Introduction to OO Analysis and Design, Berlin: Prentice Hall, 2004.</li> <li>3. Fowler, M., Patterns of Enterprise Application Architecture, Addison-Wesley, 2002.</li> <li>4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns – Elements of Reusable Object Oriented Software, Ed. Addison Wesley, 1994.</li> <li>5. Walls, Craig, Spring in Action, Third Edition, Ed. O'Reilly, 2011.</li> <li>6. Kent Beck, Test Driven Development: By Example, Ed. Addison-Wesley Professional, 2002.</li> <li>7. <a href="http://download.oracle.com/javase/tutorial/">http://download.oracle.com/javase/tutorial/</a></li> <li>8. <a href="http://msdn.microsoft.com/en-us/library/aa288436%28v=vs.71%29.aspx">http://msdn.microsoft.com/en-us/library/aa288436%28v=vs.71%29.aspx</a></li> <li>9. <a href="http://www.cs.ubbcluj.ro/~rlupsa/edu/pdp/">http://www.cs.ubbcluj.ro/~rlupsa/edu/pdp/</a></li> </ol>		



















## 9. Evaluare

Tip activitate	9.1 Criterii de evaluare <sup>4</sup>	9.2 Metode de evaluare <sup>5</sup>	9.3 Pondere din nota finală
9.4 Curs	Cunoașterea conceptelor	Examen scris	50%
	Aplicarea cunoștințelor teoretice în rezolvarea de probleme	Proiect de semestru	20%
9.5 Seminar/laborator	Aplicarea cunoștințelor teoretice în rezolvarea de probleme	Evaluarea lucrărilor de laborator	30%
9.6 Standard minim de promovare			
<ul style="list-style-type: none"> <li>• Minim 12 din 14 prezențe la laborator</li> <li>• Nota minim 5 (din 10) la examenul scris</li> <li>• Media finală minim 5 (din 10).</li> </ul>			

<sup>4</sup> Criteriile de evaluare trebuie să reflecte direct rezultatele învățării vizate la nivel de program de studii, respectiv la nivel de disciplină. Mai concret, se evaluează achizițiile de învățare menționate în rezultatele anticipate ale învățării.

<sup>5</sup> Se recomandă stabilirea atât a metodelor de evaluare finală, cât și a strategiei de evaluare pe parcurs.

## 10. Etichete ODD (Obiective de Dezvoltare Durabilă / Sustainable Development Goals)<sup>6</sup>

	<input type="radio"/>	Eticheta generală pentru Dezvoltare durabilă						
								
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
								Nu se aplică nici o etichetă
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

Data completării:

22.05.2026

Semnătura titularului de curs

Lect. dr. Radu Lușșă

Semnătura titularului de seminar

Lect. dr. Radu Lușșă

Data avizării în departament:

...

Semnătura directorului de departament

.....

<sup>6</sup> Selectați o singură etichetă, cea care, în conformitate cu [Procedura de aplicare a etichetelor ODD în procesul academic](#), se potrivește cel mai bine disciplinei. Dacă disciplina tratează tema dezvoltării durabile la modul general (de ex. prin prezentarea/introducerea cadrului general al dezvoltării durabile etc.) atunci se poate alocă eticheta generală de Dezvoltare Durabilă. Dacă niciuna dintre etichete nu descrie disciplina, selectați ultima opțiune: „Nu se aplică nici o etichetă”.