

FIȘA DISCIPLINEI

1. Date despre program

1.1. Instituția de învățământ superior	Universitatea din București
1.2. Facultatea	Facultatea de Matematică și Informatică
1.3. Departamentul	Informatică
1.4. Domeniul de studii	Informatică
1.5. Ciclu de studii	Licență
1.6. Programul de studii / Calificarea	Informatică
1.7. Forma de învățământ	Zi

2. Date despre disciplină

2.1. Denumirea disciplinei	Implementarea concurenței în limbaje de programare							
2.2. Titularul activităților de curs	Prof. Ioana Leuștean							
2.3. Titularul activităților de seminar	Asist. Bogdan Macovei							
2.4. Anul de studiu	III	2.5. Semestrul	E	2.6. Tipul de evaluare	V	2.7. Regimul disciplinei	Conținut ¹⁾	DS
							Obligativitate ²⁾	DOP

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 / proiect	2
3.4. Total ore pe semestru	56	din care: 3.5. curs	28	3.6. seminar / laborator / proiect	28
Distribuția fondului de timp					ore
3.4.1. Studiul după manual, suport de curs, bibliografie și notițe – nr. ore SI					20
3.4.2. Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					20
3.4.3. Pregătire seminare/ laboratoare/ proiecte, teme, referate, portofolii și eseuri					50
3.4.4. Examinări					4
3.4.5. Alte activități					0
3.7. Total ore studiu individual	94				
3.8. Total ore pe semestru	150				
3.9. Numărul de credite	6				

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	Cursurile de programare din programa de licență.
4.2. de competențe	Cunoștințe de programare

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	Sală de curs, dotată cu calculator, conexiune Internet și videoproiector Acces la baze de date internaționale
5.2. de desfășurare a seminarului/ laboratorului/ proiectului	Sală de laborator, dotată cu calculatoare și videoproiector

6. Competențe specifice acumulate

Competențe profesionale	Capacitate de a scrie programe în limbaje concurente. Capacitatea de a recunoaște și soluționa probleme de concurență.
Competențe transversale	Utilizarea eficientă a surselor informaționale și a resurselor de comunicare și formare profesională. Capacitatea de a citi și prelucra materiale profesionale atât în limba română cât și în limba engleză. Capacitate de a redacta și prezenta proiecte.

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

7.1. Obiectivul general al disciplinei	Studentii își vor dezvolta capacitatea de a modela și analiza probleme de concurență în diferite limbaje și paradigme de programare.
7.2. Obiectivele specifice	Capacitatea de a programa în limbaje concurente. Capacitatea de a recunoaște și de a rezolva probleme de concurență.

8. Conținuturi

8.1. Curs	Metode de predare	Observații
1. Fire de executie si mecanisme de sincronizare (Java, Python).	Pentru predare se vor folosi slideuri.	
2. Primitive de nivel înalt. Implementarea Software Transactional Memory (Haskell).		
3.Concurența folosind canale (Go).	Pentru limbajele noi se va face o scurta introducere generală.	
4. Implementarea concurenței folosing agenți (Erlang).		
5. Modele de interacțiune concurentă (în mai multe limbaje).	Cursul va pune în evidență specificul limbajelor studiate iar conceptele generale vor fi studiate comparativ.	
5. Programare asincronă (în mai multe limbaje).		
6. Implementarea unui server (în mai multe limbaje).		
Bibliografie:		
<div>[1] J. Armstrong, Programming Erlang. Software for a Concurrent World (second edition), The Pragmatic Programmers, LLC, 2013.</div> <div>[2] P. Butcher, Seven Concurrency Models in Seven Weeks. When Threads Unravel, The Pragmatic Programmers, LLC, 2014.</div> <div>[3] K. Cox-Buday, Concurrency in Go, O’Reilly, 2017.</div> <div>[4] M. Fowler. Python Concurrency with asyncio, Manning, 2022.</div> <div>[5]J.F. Gonzalez, Java 9 Concurrency Cookbook, Packt Publishing Limited, 2017.</div> <div>[6] S. Marlow, Parallel and Concurrent Programming in Haskell, O’Reilly, 2013.</div> <div>Vor fi indicate si alte resurse: articole de cercetare, documentații oficiale.</div>		
8.2. Seminar	Metode de predare-învățare	Observații
Bibliografie:		
8.3. Laborator	Metode de transmitere a informației	Observații
Laboratorul va urmări îndeaproape materia predată la curs. Vor fi implementate problemele prezentate la curs și vor fi date teme noi de laborator. Se va acorda asistență pentru realizarea proiectelor.	În laboratoare vor fi instalate limbajele predate la curs, dar se poate lucra și pe laptopul propriu.	
Bibliografie:		

8.4. Proiect	Metode de transmitere a informației	Observații
Bibliografie:		

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

Cursul este o continuare naturală a cursurilor de programare din anii anteriori. Studenții vor avea o vedere de ansamblu asupra diferitelor abordări și implementări legate de programarea concurrentă, vor putea analiza și soluționa probleme de concurență în diferite limbaje de programare.

10. Evaluare

Tip activitate	10.1. Criterii de evaluare	10.2. Metode de evaluare	10.3. Pondere din nota finală
10.4. Curs		Proiect final.	30%
10.5.1. Seminar			
10.5.2. Laborator		Teste laborator.	70%
10.5.3. Proiect			
10.6. Standard minim de performanță			

Data completării

Semnătura titularului de curs

.....

Data avizării în
departament

Director de departament

Notă:

- ¹⁾ Regimul disciplinei (conținut) - *pentru nivelul de licență se alege una din variantele:* **DF** (disciplină fundamentală) / **DS** (disciplină de specialitate) / **DC** (disciplină complementară).
- ²⁾ Regimul disciplinei (obligativitate) - *se alege una din variantele:* **DOB** (disciplină obligatorie) / **DOP** (disciplină opțională) / **DFAC** (disciplină facultativă).