

FIȘA DISCIPLINEI

Course title/ Titlul cursului:	Programare concurentă				
Course code/ Codul cursului:	Type of course/ Tipul cursului:	Level of course/ Nivelul cursului:	Year of study/ An de studiu:	Semester/ Semestru:	Number of credits/ Număr de credite:
38060253	Obligativ	Studiu de licență	3	6	6
Name of lecturer/ Numele profesorului:	Titular curs/seminar/laborator/proiect: Lector. dr. Florin Stoica				
Department/ Departament (ce coordonează disciplina):	DEPARTAMENTUL DE MATEMATICĂ ȘI INFORMATICĂ				
Extinderea disciplinei în planul de învățământ					
Lecture/Curs	Seminar	Lab/Laborator	Project/Proiect	Total (NOAD_{sem})	
24		24		48	
Bugetul de timp pentru studiu individual					
Type of activities / Denumirea activității		Hours/Nr. Ore	Type of activities / Denumirea activității		Hours/Nr. Ore
1. Studiarea notițelor de curs		15	5. Pregătirea seminariilor/laboratoarelor		10
2. Studiul după suport de curs		15	6. Elaborarea referatelor, temelor individuale		26
3. Studiarea bibliografiei minimale		21	7. Pregătirea pentru evaluările periodice		21
4. Documentare suplimentară (internet, bibliotecă ș.a.)		7	8. Participarea la consultații		5
Total ore alocate studiului individual NOSI_{sem}					120
Objectives of the course/ Obiectivele cursului:		<ul style="list-style-type: none"> a) Înțelegerea conceptelor și a tehnologiilor fundamentale de programare concurentă. b) Cunoașterea tehnicilor de programare concurentă pe platforme multiple: UNIX, Windows, Java c) Formarea deprinderilor necesare utilizării bibliotecilor standard pentru dezvoltarea de aplicații concurente (POSIX threads, clasele JAVA, API Windows). d) Manifestarea interesului pentru cunoașterea rezultatelor teoretice și aplicative recente în domeniul programării concurente. e) Adaptare la lucrul colaborativ, în echipă, pentru rezolvarea unor probleme din diverse domenii, prin implicarea în proiecte informatice în care se utilizează programarea concurentă. 			
Prerequisites/ Discipline precursoare obligatorii:		Algoritmi și structuri de date, Programare procedurală, OOP, Geometrie computațională, Tehnici de programare			
Course contents/ Conținutul cursului:		Cursul 1 - 2	Nivelele prelucrarilor concurente		

		<ul style="list-style-type: none"> a. Procesari paralele si clasificarea Flynn b. Granularitati ale paralelismului / concurenței c. Tehnica pipeline d. Procesoare vectoriale si sisteme cluster e. Paralelism si concurența la nivelul sistemului de operare f. Evaluare multiprocesor a expresiilor complexe g. Reorganizarea succesiunilor de atribuire h. Paralelizare la nivel de cicluri for
	Cursul 3 - 4	<p>Concepte abstracte utilizate in descrierea concurenței</p> <ul style="list-style-type: none"> a. Paradigme de programare nesecventiala b. Relatia procese - thread-uri c. Scheme de specificare a programelor concurente d. Situatii de exceptie generate de concurența e. Mecanisme de control al concurenței, comunicare si sincronizare f. Mecanisme de control asincron sau partial sincron g. Probleme specifice care se rezolva cu ajutorul concurenței
	Cursul 5 - 6 - 7	<p>Programare concurența la nivel de proces</p> <ul style="list-style-type: none"> a. Procese Unix, Windows, Java b. Comunicarea prin pipe intre procese c. Comunicarea intre procese folosind mecanismul de memorie partajata d. Sincronizarea proceselor folosind semafoare e. Comunicarea prin cozi de mesaje
	Cursul 8 - 9 - 10	<p>Programare concurența la nivel de thread-uri</p> <ul style="list-style-type: none"> a. Caracteristici generale b. Exemple de probleme rezolvabile prin thread-uri c. Thread-uri pe platforme Unix: Posix si Solaris d. Thread-uri pe platforme Microsoft: Windows e. Thread-uri Java
	Cursul 11 - 12	<p>Aplicatii concurente complexe</p> <ul style="list-style-type: none"> a. Scheme de proiectare a programelor concurente b. Utilizari combinate: threaduri, procese Unix, semnale c. Evaluarea unor performante ale programelor cu thread-uri
Lab/Seminary contents/ Conținutul laboratorului/seminarului:	Laborator 1	Procese Windows. Apeluri sistem sistem pentru managementul proceselor Windows.
	Laborator 2	Fire de execuție Windows. Apeluri sistem sistem pentru managementul firelor de execuție Windows.
	Laborator 3	Procese Linux. Apeluri sistem sistem pentru managementul proceselor Linux.
	Laborator 4	Fire de execuție Linux. Apeluri sistem sistem pentru managementul firelor de execuție Linux.
	Laborator 5 - 6	Mecanisme de comunicare inter-procese. Comunicația inter-procese prin pipe-uri. Comunicarea intre procese folosind

		mecanismul de memorie partajata. Comunicarea prin cozi de mesaje.
	Laborator 7	Mecanisme de sincronizare (mutex-uri, variabile de condiție, semnale). Aplicatie vizuala multi-threading Windows
	Laborator 8 - 9	Utilizarea thread-urilor in appleturi si servleturi Java
	Laborator 10 - 11	Server Java concurent pentru chat Client FTP noninteractiv
	Laborator 12	Testare finală/evaluare finală, prezentare proiecte
Teaching methods/Metode de predare:		Language of instruction/ Limba de predare: Româna
La curs se va folosi expunerea sistematică a cunoștințelor (deductivă, inductivă și formalizată, expuneri ppt); conversația frontală; conversația euristică, problematizare, studii de caz, modelarea, teme de casă. La laborator se va folosi explicația, exemplificarea, învățarea prin explorare și descoperire, verificarea, explicarea cauzelor, corectarea, stabilirea de relații funcționale, evaluarea, interpretarea, judecata critică, anticiparea, alte metode de învățare activă. Pentru curs si laborator exista suport electronic care se da studentilor la inceputul cursului. La curs se vor folosi si slide-uri si exemplificare pe calculator.		
Assesment methods/ Sisteme de evaluare:	Activități aplicative - %	1. Teme de curs/pondere= %(nCPC) 2. Referate de disciplină= %(nCPC) 3. Lucrări practice= %(CPC)
	Proiect - 60 %	CPE (CPE – condiționează participarea la examen)
	Examen parțial - %	(nCPE – nu condiționează participarea la examen)
	Examen de semestru - 40 %	(condiționează evaluarea finală)
Competențe specifice disciplinei		
1. Competențe privind cunoașterea și înțelegerea	<ul style="list-style-type: none"> - Potențialul de a valorifica la nivel de înțelegere competențele dobândite prin această disciplină. - Însușirea conceptelor și a tehnologiilor fundamentale de programare concurentă. - Abilitatea de a transpune concepte teoretice în implementări practice. 	
2. Competențe în domeniul explicării și interpretării	<ul style="list-style-type: none"> - Explicarea conceptelor și noțiunilor însușite. - Folosirea deprinderilor dobândite în punerea și rezolvarea de probleme practice. - Însușirea aspectelor teoretice privind procesele și threadurile, precum și mecanismele de sincronizare și coordonare a acestora. 	
3. Competențe instrumental - aplicative	<ul style="list-style-type: none"> - Utilizarea tehnicii de calcul pentru rezolvarea de probleme practice. - Exploatarea facilităților de lucru cu procese și threaduri oferite de către sistemele de operare Unix (Linux), Windows 7, Windows Server 2008 și de către platforma Java. 	
4. Competențe atitudinale	<ul style="list-style-type: none"> - Perspective noi privind conceptele deja cunoscute. - Participarea la manifestări științifice studentești cu proiecte dezvoltate pe baza cunoștințelor însușite - Lucrul în echipă. 	
Competențele generale sunt menționate în Fișa specializării		

Recommended reading/ Referințe bibliografice recomandate (max. 10):	<ol style="list-style-type: none">1. Stoica F., Cacovean L., Programare API, 20092. Stoica F., Sisteme de operare, Ed. Universității „Lucian Blaga”, 2007.3. F. M. Boian, C. Ferdean, R. Boian, R. Dragoș, Programare concurentă pe platforme Unix, Windows, Java, ISBN 973-650-072-1, Editura Albastră - grupul Microinformatica, Cluj, 2002.
More references/ Referințe bibliografice suplimentare:	
Regulamentul disciplinei	

Data elaborării:
10.09.2013

Titulari disciplină:
Lector. dr. Florin Stoica