

FIȘA DISCIPLINEI

UNIVERSITATEA DIN PITEȘTI

A. BENEFICIAR

Scoala Docrorala Interdisciplinara
Domeniul: Informatică

Anul universitar 2021-2024

1. Denumirea disciplinei: **Specificarea, Verificarea și Validarea formalizată a sistemelor software**

Codul disciplinei

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

2. Desfășurarea disciplinei:

Anul de stud.	Sem.		Nr.ore săptămânal				Nr. total de ore				Forme de verificare				Nr. puncte credit	
	I	II	C	S	L	P	C	S	L	P	E	C	V	P		

Șef catedră,

Numele si prenumele: Prof. univ. dr. T. Bălănescu

Semnătura :

Titular disciplină: prof. dr. Tudor Bălănescu

Cadre didactice cu activități în cadrul disciplinei (L,P): prof dr. Tudor Bălănescu

3. Obiectivele disciplinei:

- însușirea metodologiilor de specificare formalizată
- prezentarea elementelor teoretice si a tehnicilor practice de specificare, verificare , testare si depanare a sistemelor de programe secvențiale și concurente.
- înțelegerea metodelor formale de verificare , crearea deprinderii de a aplica tehnologiile de specificare și verificare pentru crearea de specificări modulare și verificabile.
- familiarizarea cu pachetele software destinate specificării și verificării automate a programelor

4. Programa analitică pentru C,S,L,P (nr.crt. / denumire temă / nr.ore.fizice)

CONȚINUTUL, PLANIFICAREA ȘI OBIECTIVELE CURSULUI

NR. CURS	NR. ORE	CONȚINUT	OBIECTIVE La examinare studenții trebuie să:
1	2	Specificarea formală ca mijloc de control al complexității sistemelor de programe	- înțelege rolul fazei de specificare din ciclul de viața al unui sistem software
2-3	4	Metodologii de specificare bazate pe logica matematică. Logica temporală. Sintaxa și semantica formulelor de logică temporală.	- fie familiarizați cu modul de specificare al sistemelor concurente și reactive
4-5	4	Logică lineară temporală (LTL); Logica temporală a arborilor de executare (CTL*) Subsetul CTL	- fie familiarizați cu modelele de timp liniar și ramificat - să fie capabili să distingă tipuri remarcabile de formule TL
6-7	4	Verificarea modelelor: model checking pentru clasele CTL și CTL*	- să fie capabili să formuleze proprietăți ale sistemelor prin formule TL și să le verifice algoritmic validitatea
7-8	4	Specificare algebrică. Tipuri abstracte de date. Teorii și modele. Algebre inițiale și algebre finale.	- să își însușească fundamentele matematice necesare pentru specificare algebrică
9-10	4	Metoda VDM (Vienna Development Method) de specificare	- să cunoască metodologia de specificare VDM și de rafinare a acestui tip de specificare - să stăpânească mecanismele de demonstrare riguroasă a proprietăților specificațiilor VDM
11-12	4	Limbajul de specificare Z.	- să poată aplica notația Z pentru specificare modulară
13-14	4	Specificarea sistemelor distribuite bazate pe evenimente	- să își însușească mecanismele de interacțiune bazată pe notificarea evenimentelor
TOTAL ORE:	28		

CONȚINUTUL, PLANIFICAREA ȘI OBIECTIVELE LABORATORULUI

NR. LAB.	NR. ORE	CONȚINUT	OBIECTIVE La examinare studenții trebuie să:
1-2	2	Exemple de specificare formală: sistem de control al unui bazin; sistem telefonic multiplex; sistem de specificare a funcționării unui lift	- dovedească înțelegerea mecanismelor abstracte de specificare,
3-4	2	Specificarea unui sistem reactiv de control la trecerea peste o cale ferată	- poată descrie sisteme prin logica temporală și să le verifice proprietățile
5	2	Leader election protocol	- să poată aplica algoritmi de validare a formulelor CTL
6	2	Specificarea structurilor stiva, coada etc. prin ADT	- aplice mecanismele algebrice de specificare abstractă
7-8	2	Specificarea VDM a unui sistem de control al unui sistem de încălzire centrală	- cunoască metodologia de definire a tipurilor și operațiilor
9-10	2	Specificarea VDM a unei rețele de calculatoare	- poată face raționamente riguroase despre proprietățile specificațiilor VDM
11-12	2	Specificarea Z a unui sistem de control al operațiilor login	- să fie familiarizati cu conceptul de schema Z
13-14	2	Specificare unui sistem producător consumator	- să distingă între proprietățile safety și liveness ale sistemelor bazate pe evenimente.
TOTAL ORE:	14		

5. Bibliografie minimă (3-5 titluri după anul 2000):

1. T. Bălănescu: Corectitudinea algoritmilor, Editura Tehnică, 1995
2. V.S. Alagar, K. Periyasamy: Specification of Software Systems, Springer, 1998
3. E.M. Clarke Jr, O. Grumberg, D. A. Peled: Model Checking, The MIT Press 2000
4. G. Mühl, L. Fiege, P. Pietzuch: Distributed Event- Based Systems, Springer, 2006

1) 6. Cunoștințe anterioare necesare: Algebră liniară, algoritmi și programare, structuri de date.

7. Evaluare:

Prezența % Activitate seminar % Activitate laborator %

Evaluare periodică % Tema de casă %

Proiect %

Alte forme de verificare periodică: _____ %

Evaluarea finală¹⁾ % [Repartizate: scris % laborator %]

- 1) 40-50% când evaluarea finală este EXAMEN
20-30% când evaluarea finală este COLOCVIU
10% când evaluarea finală este VERIFICARE

Conducator Scoala Doctorala
Numele și prenumele: Prof. univ. dr. Ing. Doru Stănescu
Semnătura:

Titular disciplină,
Numele și prenumele: Prof. univ. dr. T. Bălănescu
Semnătura:

Modele de calcul si metodologii de programare

FIȘA DISCIPLINEI

UNIVERSITATEA DIN PITEȘTI

Școala Doctorală Interdisciplinara

Domeniul: Informatică
Anul universitar 2021-2024

1. Denumirea disciplinei: **Modele de calcul si metodologii de programare**

1. Competențe specifice acumulate

Competențe profesionale	C5 Proiectarea, gestionarea ciclului de viata, integrarea si integritatea sistemelor hardware, software (4 p.c.)
	C5.1 Precizarea criteriilor relevante privind ciclul de viata, calitatea, securitatea si interactiunea sistemului de calcul cu mediul si cu operatorul uman (1 p.c.)
	C5.2 Utilizarea unor cunostinte interdisciplinare pentru adaptarea sistemului informatic în raport cu cerintele domeniului de aplicatii (1 p.c.)
	C5.3 Utilizarea unor principii si metode de baza pentru asigurarea securitatii, sigurantei si usurintei în exploatarea a sistemelor de calcul (1 p.c.)
	C5.4 Utilizarea adecvata a standardelor de calitate, siguranta si securitate în prelucrarea informatiilor (1 p.c.)

Competențe transversale	<p>CT1 Comportarea onorabila, responsabila, etica, în spiritul legii pentru a asigura reputatia profesiei.</p> <p>CT3 Demonstrarea spiritului de initiativa si actiune pentru actualizarea cunostintelor profesionale, economice si de cultura organizationala.</p>
-------------------------	---

2. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	- Cunoașterea conceptelor legate de modelarea sistemelor de programe, verificarea proprietăților modelelor și a protocoalelor de comunicare. Înșușirea abilităților de elaborare a modelelor, de clasificare a proprietăților acestora și de exprimare formalizată a proprietăților. Familiarizarea cu tehnicile de verificare și testare automatizată.
7.2 Obiectivele specifice	<ul style="list-style-type: none"> - Cunoașterea metodologiilor de demonstrare formală a proprietăților programelor cu instrucțiuni iterative. - Înșușirea tehnicilor de specificare a sistemelor de programe utilizând sisteme de tranziție. - Cunoașterea criteriilor de clasificare a proprietăților sistemelor. - Familiarizarea cu tehnicile de verificare a modelelor (model checking) - Utilizarea metodologiilor formalizate de elaborare a testelor și de testare automatizată. - Laboratorul este axat pe utilizarea sistemelor specializate în verificarea și testarea automată.

3. Conținuturi

8.1. Curs		Metode de predare	Observații Resurse folosite
1	<p>1. Introducere</p> <p>1.1. Formal și informal în specificare, verificare și testare</p> <p>1.2. Verificare vs. Validare</p> <p>1.3. Verificare statică vs. Verificare dinamică</p> <p>1.4. Sisteme deductive pentru demonstrarea proprietăților programelor</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri	Calculator, Videoproiector, Suport documentar
2	<p>2. Sisteme deductive pentru instrucțiuni iterative</p> <p>2.1. Corectitudine parțială și corectitudine totală</p> <p>2.2. Proprietăți invariante și funcții de terminare</p> <p>2.3. Modelarea calculului paralel cu instrucțiuni nedeterminate</p> <p>2.4. Sisteme deductive pentru verificarea statică a programelor secvențiale și a celor paralele</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
3	<p>3. Specificarea formalizată prin sisteme de tranziție</p> <p>3.1. Sisteme de tranziție cu număr infinit de stări</p> <p>3.2. Modelarea procesării paralele prin întretesere</p> <p>3.3. Specificarea protocoalelor de comunicare prin sisteme de grafuri program</p> <p>3.4. Calcul asincron și calcul sincron. Sincronizare prin handshaking.</p> <p>3.5. Comunicare prin canale</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
4	<p>4. Verificarea formalizată a modelelor sistemelor de programe</p> <p>3.1. Clasificarea proprietăților sistemelor. Proprietăți safety și proprietăți liveness</p> <p>3.2. Criterii de executare imparțială a proceselor. Fairness. Imparțialitate slabă și imparțialitate tare.</p> <p>3.3. Logică temporală. Specificarea proprietăților utilizând formule LTL sau formule CTL</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
5	<p>5. Sisteme specializate de verificare</p> <p>1.1. SPIN</p> <p>2.1. UPPAAL</p> <p>3.1. NuSMV</p> <p>4.1. Limbajul Promela</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
6	<p>6. Specificare prin mașini cu număr finit de stări</p> <p>6.1. Testare vs. demonstrare formală</p> <p>6.2. Specificare prin mașini cu număr finit de stări</p> <p>6.3. Relația de similaritate și disimilaritate între stări. Calculul algoritmic al relațiilor</p> <p>6.4. Mașini cu număr minim de stări</p> <p style="text-align: right;">-Timp alocat 4 ore</p>	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
7	<p>7. Testare automată</p> <p>7.1. Metoda W de elaborare a testelor</p>	Prelegere Dezbateri	Calculator, Videoproiector,

	7.2. Metoda Wp de elaborare automată a testelor. 7.3. Implementarea sistemelor de testare automată -Timp alocat 4 ore	Studiu de caz	Suport documentar
Bibliografie: 1. V.S. Alagar, K. Periyasamy: Specification of Software Systems, Springer, 1998 2. T. Bălănescu: Corectitudinea algoritmilor, Editura Tehnică, 1995 3. E.M. Clarke Jr, O. Grumberg, D. A. Peled: Model Checking, The MIT Press, 2000 4. M. Holcombe, F. Ipate: Correct Systems: building business process solutions, Springer Verlag, 1998. 5. Aditya P. Mathur: Foundation of Software Testing, Pearson Education, 2008 6. G. Mühl, L. Fiege, P. Pietzuch: Distributed Event- Based Systems, Springer, 2006			
8.2. Aplicații – Laborator		Metode de predare	Observații Resurse folosite
1	1. Introducere 1.1. Verificarea statică a algoritmilor iterativi 1.2. Proprietăți invariante și funcții de terminare -Timp alocat 2 ore	Prelegere Dezbateri	Calculator, Videoproiector, Suport documentar
2	2. Sisteme deductive pentru instrucțiuni iterative 1.3. Verificarea/testarea unor algoritmi de sortare 1.4. Verificarea/testarea unor algoritmi de căutare -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
3	3. Specificarea formalizată prin sisteme de tranziție 3.1. Specificarea unui sistem automat de trecere la nivel de cale ferată 3.2. Specificarea protocoalelor de sincronizare cu semafoare 3.3. Protocolul lui Petterson -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
4	4. Verificarea formalizată a modelelor sistemelor de programe 4.1. Proprietăți safety și proprietăți liveness ale sistemelor utilizând formule LTL sau formule CTL 4.2. Programare in Promela 4.3. Verificarea modelelor cu sistemul SPIN -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
5	5. Sisteme specificate prin automate cu cronometru 5.1. Variantă cu cronometru a sistemului de trecere la calea ferată 5.2. Verificarea acestui model utilizând utilitarul UPPAAL -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
6	6. Specificare prin mașini cu număr finit de stări 6.1. Specificare prin mașini cu număr finit de stări a unor protocoale de comunicare 6.2. Implementarea și testarea acestor sisteme -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
7	7. Testare automată 7.1. Metoda W de elaborare a testelor 7.2. Metoda Wp de elaborare automată a testelor. 7.3. Implementarea sistemelor de testare automată -Timp alocat 2 ore	Prelegere Dezbateri Studiu de caz	Calculator, Videoproiector, Suport documentar
Bibliografie: 1. V.S. Alagar, K. Periyasamy: Specification of Software Systems, Springer, 1998 2. T. Bălănescu: Corectitudinea algoritmilor, Editura Tehnică, 1995 3. E.M. Clarke Jr, O. Grumberg, D. A. Peled: Model Checking, The MIT Press, 2000 4. M. Holcombe, F. Ipate: Correct Systems: building business process solutions, Springer Verlag, 1998 5. Aditya P. Mathur: Foundation of Software Testing, Pearson Education, 2008 6. G. Mühl, L. Fiege, P. Pietzuch: Distributed Event- Based Systems, Springer, 2006			

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

Competențele dobândite la disciplină permit absolvenților să coordoneze echipele de specificare, implementare și validare a sistemelor software complexe de natură paralelă, concurrentă sau distribuită. Utilizarea cunoștințelor dobândite contribuie la diminuarea costurilor de realizare a unor astfel de sisteme software prin: specificarea formalizată; elaborarea sistematică a datelor de test; utilizarea unor sisteme software de testare automată; complementarea activității de testare prin includerea acțiunilor de verificare a proprietăților modelelor.

5. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	a) Prezență Curs b) Examen	a) Înregistrare prezență curs b) Scris - verificare cunoștințe teoretice	10% 50%

5. Bibliografie minimala:

- M. Huth, M. Ryan: Logic in Computer Science: Modelling and Reasoning about Systems. Cambridge University Press, Second Edition 2004. ISBN: 0 521 54310.
- M. Holcombe, F. Ipaté: Correct Systems: building business process solutions, Springer Verlag, 1998.
- D. MacKenzie: Mechanizing Proof: computing, risk and trust. MIT Press, 2001.
- J. Spivey, The Z Notation: a Reference Manual, Prentice Hall, 1989.

6. Cunoștințe anterioare necesare: Elemente fundamentale de programare orientată pe obiecte

7. Evaluare:

Prezența	<input type="text" value="1"/> <input type="text" value="0"/>	%	Activitate seminar	<input type="text" value="-"/> <input type="text" value="-"/>	%	Activitate laborator	<input type="text" value="3"/> <input type="text" value="0"/>	%	
Evaluare periodică	<input type="text" value="-"/> <input type="text" value="-"/>	%	Tema de casă	<input type="text" value="-"/> <input type="text" value="-"/>	%				
Proiect	<input type="text" value="2"/> <input type="text" value="0"/>	%							
Alte forme de verificare periodică:	_____						<input type="text" value="-"/> <input type="text" value="-"/>	%	
Evaluarea finală ¹⁾	<input type="text" value="4"/> <input type="text" value="0"/>	%	[Repartizate:	scris	<input type="text" value="2"/> <input type="text" value="0"/>	%	laborator	<input type="text" value="2"/> <input type="text" value="0"/>	%]

- 1) 40-50% când evaluarea finală este EXAMEN
20-30% când evaluarea finală este COLOCVIU
10% când evaluarea finală este VERIFICARE

Șef catedră,
Numele și prenumele: Prof. univ. dr. Ing. Doru Stănescu
Semnătura:

Titular disciplină,
Numele și prenumele: Prof. univ. dr. T. Bălănescu
Semnătura: