

FIȘA DISCIPLINEI

Paradigme de programare

Anul universitar 2017-2018

1. Date despre program

1.1	Instituția de învățământ superior	Universitatea din Pitești
1.2	Facultatea	Electronică, Comunicații și Calculatoare
1.3	Departamentul	Electronică, Calculatoare și Inginerie Electrică
1.4	Domeniul de studii	Calculatoare și tehnologia informației
1.5	Ciclul de studii	Licență
1.6	Programul de studii / Calificarea	Calculatoare / Inginer calculatoare

2. Date despre disciplină

2.1 Denumirea disciplinei						Paradigme de programare					
2.2 Titularul activităților de curs						conf. univ. dr. Doru CONSTANTIN					
2.3 Titularul activităților de laborator						conf. univ. dr. Doru CONSTANTIN					
2.4	Anul de studii	II	2.5	Semestrul	II	2.6	Tipul de evaluare	E	2.7	Regimul disciplinei	D/A

3. Timpul total estimat

3.1	Număr de ore pe săptămână	3	3.2	din care curs	2	3.3	laborator	1
3.4	Total ore din planul de învățământ	42	3.5	din care curs	28	3.6	laborator	14
Distribuția fondului de timp								ore
Studiul după manual, suport de curs, bibliografie și notițe								14
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren								16
Pregătire seminarii/laboratoare, teme, referate, portofolii, eseuri								16
Tutorat								2
Examinări								6
Alte activități								-
		3.7	Total ore studiu individual			54		
		3.8	Total ore pe semestru			96		
		3.9	Număr de credite			4		

4. Precondiții (acolo unde este cazul)

4.1	De curriculum	Parcursirea și/sau promovarea următoarelor discipline: structuri de date, programarea calculatoarelor, analiza algoritmilor
4.2	De competențe	abilități de sintetizare și raționament logic

5. Condiții (acolo unde este cazul)

5.1	De desfășurare a cursului	Sală cu o capacitate de minim 100 locuri dotată cu videoproiector și ecran de proiecție, 2 table.
5.2	De desfășurare a laboratorului/proiectului	Sala de laborator dotată cu calculatoare, capacitate maximă 18 studenți/laborator.

6. Competențe specifice acumulate

Competențe profesionale	C6.1 Descrierea componentelor sistemelor inteligente; C6.2 Utilizarea de instrumente specifice domeniului pentru explicarea funcționării sistemelor inteligente; C6.3 Aplicarea principiilor și metodelor de baza pentru specificarea de soluții la probleme tipice utilizând sisteme inteligente; C6.5 Dezvoltarea și implementarea de proiecte profesionale pentru sisteme inteligente.
Competențe transversale	-

7. Obiectivele disciplinei

7.1 Obiectivul general al disciplinei	- Cunoașterea modului în care principalele concepte aferente programării sunt materializate și utilizate în raport cu paradigma de programare aleasă. - Redarea comparativă a elementelor esențiale ale unor limbaje de programare funcțională, asociativă și logică și aplicarea unor tehnici de programare specifice limbajelor studiate.
7.2 Obiectivele specifice	<i>Obiective cognitive:</i> ■ Cunoașterea principiilor ce caracterizează modelele de calculabilitate, a diverselor perspective conceptuale asupra noțiunii de calculabilitate efectivă; ■ Cunoașterea și înțelegerea conceptelor și a elementelor de paradigme de programare și analiza influenței perspectivei alese asupra procesului de modelare și rezolvare a problemelor. <i>Obiective procedurale:</i> ■ Formarea deprinderilor și abilitatea de a utiliza un mediu de programare logică; ■ Formarea deprinderilor și abilitatea de a utiliza elementele unui limbaj de programare, cunoașterea mecanismelor expresive aferente paradigmei, cu accent pe aspectul comparativ. ■ <i>Obiective atitudinale:</i> Rigurozitate în proiectarea și implementarea algoritmilor de programare logică.

8. Conținuturi

8.1. Curs		Nr. de ore	Metode de predare	Observații Resurse folosite
1	Introducere și calculul lambda.	4	Prelegere Dezbateri Explicația Descrierea Exemplificarea Problematizarea Exercițiul	Calculator, Videoproiector
2	Programare funcțională în Racket.	4		
3	Programare funcțională în Haskell.	4		
4	Clase în Haskell și recapitulare paradigme funcționale.	4		
5	Introducere în Prolog și logica propozițională.	4		
6	Logica cu predicate de ordinul întâi.	4		
7	Programare logică în Prolog.	4		

Bibliografie

1. Note de curs și laborator – suport electronic – Doru Constantin.
2. L. State: *Introducere în programarea logică*, Editura Fundației România de mâine, București, 2004.
3. I. Bratko: *PROLOG*, Addison Wesley, 2001.
4. J. Hughes, S.P. Jones, P. Wadler, "A history of Haskell: being lazy with class", ACM SIGPLAN Conf. on History of Programming Languages. San Diego, California, 2007; L. Sterling, Shapiro: *The Art of Prolog*, MIT Press, 2001.
5. F. Matthew and PLT. Reference: *Racket*. PLT Inc., PLT-TR-2010-1, 2010. <http://racket-lang.org/tr1/>.
6. J.W. Lloyd: *Foundations of Logic Programming*, Springer Verlag, 1987.
7. K.R. Prashanth, S. Tobin-Hochstadt, "Functional data structures for Typed Racket", In Proc. Works. Scheme and Functional Programming, pp. 1–7, 2010.

8.2. Aplicații – Laborator		Nr. de ore	Metode de predare	Observații Resurse folosite
1	Aplicații introductive în limbajul funcțional Racket	2	Studiul de caz Lucrul în grup Problematizarea Exercițiul Dezbateri	Calculator, Videoproiector
2	8. Aplicații în limbajul funcțional Racket - recursivitate	2		
3	Aplicații în limbajul funcțional introductive în limbajul funcțional Haskell	2		
4	Aplicații Haskell: polimorfism și clase.	2		
5	Limbajul Prolog: mediul de programare, sintaxa, predicate predefinite;	2		
6	Instruire Prolog: aplicații cu algoritmi numerici și nenumeri.	2		
7	Aplicații cu arbori și grafuri;	2		
	Dezvoltarea și implementarea unor aplicații cu grad de dificultate avansat (metode de sortare, definirea și utilizarea structurilor de date (prelucrări asupra listelor, listelor de liste, etc.), problema celor opt regine, etc.).	4		

Bibliografie

1. Note de curs și laborator – suport electronic – Doru Constantin.
2. L. State: *Introducere în programarea logică*, Editura Fundației România de mâine, București, 2004.
3. I. Bratko: *PROLOG*, Addison Wesley, 2001.
4. J. Hughes, S.P. Jones, P. Wadler, "A history of Haskell: being lazy with class", ACM SIGPLAN Conf. on History of Programming Languages. San Diego, California, 2007; L. Sterling, Shapiro: *The Art of Prolog*, MIT Press, 2001.
5. F. Matthew and PLT. Reference: *Racket*. PLT Inc., PLT-TR-2010-1, 2010. <http://racket-lang.org/tr1/>.
6. J.W. Lloyd: *Foundations of Logic Programming*, Springer Verlag, 1987.
7. K.R. Prashanth, S. Tobin-Hochstadt, "Functional data structures for Typed Racket", In Proc. Works. Scheme and Functional Programming, pp. 1–7, 2010.

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

Coroborarea conținuturilor disciplinei cu așteptările comunității epistemice se realizează prin stabilirea de întâlniri cu principalii actori de pe piața IT locală (RoWeb, Lisa, Proding, Kepler, Osf, Endava, etc.), precum și prin vizite la firmele de profil și schimburi de bune practici cu cadre didactice din alte universități.

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 Metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	a) examen	a) scris - verificare cunoștințe teoretice și a elemente de implementare	50%
10.5 Laborator	a) activitate laborator și teme de casă b) rezolvarea problemelor folosind cunoștințele cumulate la laborator și curs	a) evaluarea activității și a temelor b) evaluare periodică privind implementări în limbajul logic studiat	10% 40%
10.6 Standard minim de performanță	*Notă minimă 5 la activitățile de laborator și notă minimă 5 la fiecare din subiectele de la examenul final. *Set de cunoștințe minimale pentru promovarea examenului final: cunoașterea principiilor de bază ale unui limbaj de programare logică; cunoașterea principalelor elemente de fundamentare matematică a unui limbaj de programare logică; cunoașterea elementelor de bază, implementări ale unor algoritmi de bază.		

Data completării
19.09.2017

Titular de curs și proiect
Conf.univ.dr. Doru CONSTANTIN

Titular de laborator
Conf.univ.dr. Doru CONSTANTIN

Data avizării în Departament
25.09.2017

Director Departament (prestator)
Conf.univ.dr. Doru CONSTANTIN

Director Departament (beneficiar)
Prof.univ.dr.ing. Gheorghe ȘERBAN