

FIȘA DISCIPLINEI

Programare logică și programare funcțională 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. Date despre disciplina												
2.1	Denumirea disciplinei					Programare logică și programare funcțională						
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	cunoștințe de proiectare a 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 bază 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 conceptelor legate de programarea logică, de structură și mecanismele interne ale unui limbaj de programare logică; - Proiectarea de sisteme inteligente bazate pe aplicarea tehnicilor de bază din programarea declarativă și funcțională, aplicații în inteligența artificială.
7.2 Obiectivele specifice	<i>Obiective cognitive:</i> ► Cunoașterea principiilor fundamentale care guvernează programarea logică și funcțională; ► Cunoașterea fundamentelor matematice ce definesc limbajul de programare și implementările informatice prin descrierea conceptelor de bază ale limbajelor de programare logică și funcțională; ► Cunoașterea conceptelor, elementelor, caracteristicilor unui limbaj nestructurat bazat pe principii de

	<p>matching și mecanisme de deducție logică ce referă principiul de backtracking recursiv.</p> <p>Obiective procedurale:</p> <ul style="list-style-type: none"> ► Formarea deprinderilor și abilitatea de a utiliza un mediu de programare logică; ► Formarea deprinderilor și abilitatea de a realiza programe într-un limbaj de programare logică, precum și capacitatea de implementare și dezvoltare a unor aplicații cu grad de dificultate avansat. <p>Obiective atitudinale:</p> <ul style="list-style-type: none"> ► Rigurozitate în proiectarea și implementarea algoritmilor de programare logică.
--	---

8. Conținuturi

8.1. Curs		Nr. ore	Metode de predare	Observații Resurse folosite
1	Sintaxa și semantica limbajului de calcul propozițional logic;	2	Prelegere Dezbateri Explicația Descrierea Exemplificarea Problematizarea Exercițiul	Calculator Videoproiector
2	Sintaxa limbajelor de ordinul I (definirea limbajului, a structurilor de interes de termeni, atomi și formule, arbori de structură);	2		
3	Definirea mulțimilor de variabile libere și legate și a substituțiilor asupra structurilor de termeni, atomi și formule;	2		
4	Algoritmul Robinson de unificare a expresiilor;	2		
5	Semantici pentru limbajele de ordin I – interpretarea termenilor, a atomilor și a formulelor;	2		
6	Validabilitatea mulțimilor de formule și modele;	2		
7	Transformarea programelor logice; forme normale prenex;	2		
8	Forme normale Skolem, lema de normalizare și reprezentări clauzale;	2		
9	Arbori semantici; teorema Herbrand, modele minimale;	2		
10	Metoda rezolutivă pentru verificarea validabilității/invalidității mulțimilor finite de clauze;	2		
11	Demonstrarea automată bazată pe principiul rezoluției;	2		
12	Teorema de consistență-completitudine a rezoluției pentru limbajele de ordinul I;	2		
13	Programare logică cu restricții. Programare logică inductivă. Elemente de programare funcțională.	4		

Bibliografie:

1. Note de curs și laborator - suport electronic - Doru Constantin;
2. L. State: Introducere în programarea logică, Editura Fundației României de mâine, București, 2004;
3. J.W. Lloyd: Foundations of Logic Programming, Springer Verlag, 1987;
4. L. Sterling, Shapiro: The Art of Prolog, MIT Press, 2001; I. Bratko: PROLOG, Addison Wesley, 2001;
5. C. Cocianu, L. State: Tehnici de clasificare cu apl. în analiza datelor document, Editura ASE București, 2008;
6. N. L. Tudor: Programare logică și sisteme expert. Aplicații Visual Prolog, Matrix Rom, 2012.

8.2. Aplicații – Laborator		Nr. ore	Metode de predare	Observații Resurse folosite
1	Prezentarea limbajului de programare logică Prolog, SWI Prolog: mediul de programare, sintaxa;	2	Studiul de caz Lucrul în grup Problematizarea Exercițiul Dezbateri	Calculator cu soft Prolog
2	Aplicații aritmetice în limbajul Prolog;	2		
3	Definirea listelor și predicate standard de prelucrare a listelor în Prolog;	2		
4	Aplicații de aprofundare privind prelucrarea listelor în Prolog;	2		
5	Definirea structurilor în Prolog și aplicații;	1		
6	Definirea structurilor de arbori și grafuri cu aplicații în Prolog;	2		
7	Verificarea periodică a cunoștințelor acumulate;	1		
8	Dezvoltarea și implementarea unor aplicații cu grad de dificultate avansat - metode de sortare;	1		
9	Dezvoltarea și implementarea unor aplicații cu grad de dificultate avansat - utilizarea structurilor de date (prel. avansate asupra listelor, listelor de liste), problema celor opt regine, implementarea unui automat finit determinist, aplicații de programare funcțională.	1		

Bibliografie:

1. Note de curs și laborator – suport electronic – Doru Constantin;
2. L. State: Introducere în programarea logică, Editura Fundației României de mâine, București, 2004;
3. J.W. Lloyd: Foundations of Logic Programming, Springer Verlag, 1987;
4. L. Sterling, Shapiro: The Art of Prolog, MIT Press, 2001; I. Bratko: PROLOG, Addison Wesley, 2001;
5. N. L. Tudor: Programare logică și sisteme expert. Aplicații Visual Prolog și Exsys, Matrix Rom, 2012.

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	* Nota minimă 5 la activitățile de laborator și nota 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