

FIȘA DISCIPLINEI
CALCUL INTELIGENT- MODELE DE INSPIRATIE BIOLOGICA

1. Date despre program

1.1. Instituția de învățământ	Universitatea „1 Decembrie 1918” din Alba Iulia
1.2. Facultatea	de Informatica și Inginerie
1.3. Departamentul	de Informatica, Matematica si Electronica
1.4. Domeniul de studii	Informatică
1.5. Ciclul de studii	Licență
1.6. Programul de studii	Informatică Cod COR: Analist/251201, Programator de sistem informatic/251204, Inginer de sistem în informatică/251203 ESCO-08: 2511/ Systems Analyst, 2512/ Software developers

2. Date despre disciplină

2.1. Denumirea disciplinei	<i>Calcul inteligent- modele de inspiratie biologica</i>	2.2. Cod disciplină	INFO313
2.3. Titularul activității de curs	Rotar Corina		
2.4. Titularul activității de seminar/lab	Rotar Corina/ Onita Nagy Daniela		
2.5. Anul de studiu	III	2.6. Semestrul	II
		2.7. Tipul de evaluare (E/C/VP)	C
		2.8. Regimul disciplinei (O – obligatorie, Op – opțională, F – facultativă)	Op

3. Timpul total estimat

3.1. Numar ore pe saptamana	2	din care: 3.2. curs	2	3.3. seminar/laborator	3
3.4. Total ore din planul de învățământ	60	din care: 3.5. curs	24	3.6. seminar/laborator	36
Distribuția fondului de timp					ore
Studiul după manual, suport de curs, bibliografie și notițe					24
Documentare suplimentară în bibliotecă, pe platformele electronice de specialitate și pe teren					24
Pregătire seminarii/laboratoare, teme, referate, portofolii și eseuri					34
Tutoriat					-
Examinări					8
Alte activități					-

3.7 Total ore studiu individual	90
3.8 Total ore din planul de învățământ	60
3.9 Total ore pe semestru	150
3.10 Numărul de credite	6

4. Precondiții (acolo unde este cazul)

4.1. de curriculum	1. Fundamentele programarii
4.2. de competențe	- C1.1 Descrierea adecvată a - paradigmelor de programare și a mecanismelor de limbaj specifice, precum și identificarea diferenței dintre aspectele de ordin semantic și sintactic - C1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date.

	- C 1.4 Testarea unor aplicații pe baza unor planuri de test.
	- C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente.

5. Condiții (acolo unde este cazul)

5.1. de desfășurare a cursului	<i>Sala dotata cu videoproiector/tabla</i>
5.2. de desfășurarea a seminarului/laboratorului	<i>Laboratoare – calculatoare dotate cu: Visual Studio min. 2010, acces Internet.</i>

6. Competențe specifice acumulate

Competențe profesionale	<p>C1. Programarea în limbaje de nivel înalt <i>C1.3 Elaborarea codurilor sursă adecvate și testarea unitară a unor componente într-un limbaj de programare cunoscut, pe baza unor specificații de proiectare date.</i> <i>C 1.4 Testarea unor aplicații pe baza unor planuri de test.</i> <i>C1.5 Dezvoltarea de unități de program și elaborarea documentațiilor aferente.</i></p> <p>C3.Utilizarea instrumentelor informatice in context interdisciplinar <i>C3.1. Descrierea de concepte, teorii si modele folosite in domeniul de aplicare.</i> <i>C3.2 Identificarea si explicarea modelelor informatice de baza adecvate domeniului de aplicare.</i> <i>C3.3. Utilizarea modelelor si instrumentelor informatice si matematice pentru rezolvarea problemelor specifice domeniului de aplicare.</i> <i>C3.4. Analiza datelor si a modelelor</i></p> <p>C4. Utilizarea bazelor teoretice ale informaticii si a modelelor formale <i>C4.3 Identificarea modelelor si metodelor adecvate pentru rezolvarea unor probleme reale.</i> <i>C4.4 Utilizarea simulării pentru studiul comportamentului modelelor realizate si evaluarea performantelor.</i></p>
Competențe transversale	Nu e cazul

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

7.1 Obiectivul general al disciplinei	<i>Dezvoltarea capacității studentului de a dezvolta aplicații software dedicate rezolvării problemelor de complexitate medie-mare exploatând paradigma calculului inteligent.si principiile algoritmilor evolutivi. Dezvoltarea abilitatii studentului de a găsi metode neconvenționale de rezolvare a problemelor..</i>
7.2 Obiectivele specifice	<i>Studierea algoritmilor dezvoltați pe baza paradigmei naturale. Formarea deprinderilor de a aborda probleme de complexitate ridicată din perspectiva algoritmilor evolutivi. Studiu analitic al avantajelor și dezavantajelor algoritmilor stochastici versus algoritmi tradiționali în rezolvarea problemelor de optimizare.</i>

8. Conținuturi

8.1 Curs	Metode de predare	Observații
1. Noțiuni introductive	<i>Prelegere, discutii</i>	
2. Paradigma Algoritmilor Genetici	<i>Prelegere, discutii</i>	
3. Paradigma Strategiilor Evolutive	<i>Prelegere, discutii</i>	
4. Programarea Genetică. Programarea Evolutivă	<i>Prelegere, discutii</i>	
5. Sisteme imune artificiale	<i>Prelegere, discutii</i>	
6. Tehnica Particle Swarm Optimization	<i>Prelegere, discutii</i>	
7. Colonii de furnici. Alte paradigme naturale	<i>Prelegere, discutii</i>	

8. Aplicații ale algoritmilor evolutivi în optimizare	<i>Prelegere, discutii</i>	
9. Introducere în logica fuzzy. Sisteme fuzzy.	<i>Prelegere, discutii</i>	
10. Introducere în calculul neural	<i>Prelegere, discutii</i>	
8.2 Bibliografie		
1. Rotar C., Modele naturale si algoritmi evolutivi, Ed. Accent, Cluj Napoca, 2008. 2. Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., 1989. 3. Bäck T., Evolutionary Algorithms in Theory and Practice, Oxford University Press, 1996 4. Dumitrescu D., Lazzerini B., Jain L.C., Dumitrescu A., Evolutionary Computation, CRC Press, Boca Raton London, New York, Washington D.C., 2000		
Seminar-laborator		
1. Notiuni introductive	Lucrare practica de laborator	
2. Paradigma Algoritmilor Genetici. Descrierea algoritmului genetic. Functia de evaluare. Operatori specifici: <i>crossover</i> , <i>mutation</i> , <i>selection</i> .	Lucrare practica de laborator	
3. Paradigma Strategiilor Evolutive. Operatori specifici.	Lucrare practica de laborator	
4. Programarea Genetică. Exemple de aplicatii	Lucrare practica de laborator	
5. Programarea Evolutivă. Exemple de aplicatii	Lucrare practica de laborator	
6. Sisteme imune artificiale. Exemple de aplicatii (Securitatea retelelor).	Lucrare practica de laborator	
7. Tehnica Particle Swarm Optimization. Exemple de aplicatii	Lucrare practica de laborator	
8. Colonii de furnici. Problema Comis Voiajorului in abordarea evolutiva. Alte paradigme naturale.	Lucrare practica de laborator	
9. Aplicații ale algoritmilor evolutivi în optimizare. Optimizare multimodala, multicriteriala, dinamica. - 2 laboratoare	Lucrare practica de laborator	
Bibliografie		
1. Rotar C., Modele naturale si algoritmi evolutivi, Ed. Accent, Cluj Napoca, 2008. 2. Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., 1989. 3. Bäck T., Evolutionary Algorithms in Theory and Practice, Oxford University Press, 1996 4. Dumitrescu D., Lazzerini B., Jain L.C., Dumitrescu A., Evolutionary Computation, CRC Press, Boca Raton London, New York, Washington D.C., 2000		

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

- *La ora actuală se înregistrează un interes accentuat înspre dezvoltarea de aplicații software inteligente în variate domenii precum telefonie mobilă, industria jocurilor, etc. Disciplina Calcul inteligent vine în sprijinul formării de specialiști în această direcție, formînd deprinderea de a aplica strategii și algoritmi inteligenți acolo unde metodele tradiționale nu sunt eficiente.*
- *Coexistența în cadrul Universității specializarilor tehnice, în particular al specializării Electronică aplicată, este un motiv suplimentar pentru încurajarea specilizării studentului în direcția Inteligenței artificiale prin formarea de echipe interdisciplinare și complementare.*

10. Evaluare

Tip activitate	10.1 Criterii de evaluare	10.2 metode de evaluare	10.3 Pondere din nota finală
10.4 Curs	<i>Evaluare finala</i>	<i>Examen oral</i>	60%
	-	-	-
10.5 Seminar/laborator	<i>Verificare pe parcurs</i>	<i>Proiect individual</i>	40%
	-	-	-
10.6 Standard minim de performanță:			
Implementarea și documentarea de unități de program în limbaje de programare de nivel înalt și folosirea eficientă a mediilor de programare. Implementarea și aplicarea unui algoritm evolutiv în cazul unei probleme reale.			

Observatii: Recuperarea laboratoarelor se poate face în regim de consultații în timpul semestrului. De asemenea, în cazuri bine motivate, recuperarea orelor de laborator se mai poate face prin prezentarea de către student a portofoliului complet de lucrări practice - în ultima săptămână din semestrul II, în orele de consultații ale cadrului didactic titular.

Data completării

.....

Semnătura titularului de curs

.....

Semnătura titularului de seminar

.....

Data avizării în departament

.....

Semnătura director departament

.....

Data avizării în Consiliul Facultății

.....

Semnătura decan

.....