SYLLABUS

COMPUTATIONAL INTELLIGENCE

1. Information on academic programme

1.1. University	„1 Decembrie 1918" from Alba Iulia
1.2. Faculty	Faculty Of Exact Sciences and Engineering
1.3. Department	Informatics, Mathematics and Electronics Department
1.4. Field of Study	Computer Science
1.5. Cycle of Study	Undergraduate
1.6. Academic programme / Qualification	Computer Science / 251201, 251203, 251204

2. Information of Course Matter

2.1. **Course**	*COMPUTATIONAL INTELLIGENCE*			2.2. **Code**		CSE 302	
2.3. **Course Leader**		Rotar Corina					
2.4. **Seminar Tutor**		Rotar Corina					
2.5. Academic Year	**III**	2.6. Semester	**II**	2.7. Type of Evaluation (E – final exam/ CE - colloquy examination / CA -continuous assessment)	**E**	2.8. Type of course (**C**– Compulsory, **Op** – optional, **F** - Facultative)	**Op**

3. Course Structure (Weekly number of hours)

3.1. Weekly number of hours	**4**	3.2. course	***2***	3.3. seminar, laboratory	***2***
3.4. Total number of hours in the curriculum	**48**	3.5. course	***24***	3.6. seminar, laboratory	***24***

Allocation of time:	Hours
Individual study of readers	***20***
Documentation (library)	***20***
Home assignments, Essays, Portfolios	***50***
Tutorials	**-**
Assessment (examinations)	***12***
Other activities…….	**-**

3.7 Total number of hours for individual study	**102**
3.9 Total number of hours per semester	**150**
3.10 Number of ECTS	**6**

4. Prerequisites (*where applicable*)

4.1. curriculum-based	Imperative / Procedural programming
4.2. competence-based	C1.1 The appropriate description of programming paradigms and of specific language mechanisms, as well as the identification of differences between semantic and syntactic aspects. C1.3 The development of correct source codes and the testing of various components in a known programming language, given a set of design specifications

5. Requisites (*where applicable*)

5.1. course-related	***Room equipped with video projector / boar***
5.2. seminar/laboratory-based	***Laboratory – computer, Software: Visual Studio 2010, Internet access.***

6. Specific competences to be aquired (chosen by the course leader from the programme general competences grid)

Professional competences	***C1 Programming in high-level languages*** ***C3 The use of computer tools in an interdisciplinary context*** ***C4 The use of the theoretical basis of computer science and of formal models***
Transversal competences	Not applicable

7. Course objectives (as per the programme specific competences grid)

7.1 General objectives of the course	Develop the students' ability to design software that is dedicated for solving the difficult problems by exploiting evolutionary/bio-inspired algorithms.
7.2 Specific objectives of the course	Study of the algorithms that is based on natural paradigms. Skills for approaching the complex problems in terms of evolutionary algorithms. Analytical study of the advantages and disadvantages of traditional algorithms versus stochastic algorithms for optimization problems.

8. Course contents

8.1 Course (learning units)	**Teaching methods**	**Remarks**
1. Fundamentals of Intelligence Computation	***Lecture, conversation, exemplification***	**2**
2. Paradigm of Genetic Algorithms	***Lecture, conversation, exemplification***	**2**
3. Paradigm of Evolutionary Strategies	***Lecture, conversation, exemplification***	**2**
4. Genetic Programming. Evolutionary programming	***Lecture, conversation, exemplification***	**2**
5. Artificial Immune Systems	***Lecture, conversation, exemplification***	**2**
6. Particle Swarm Optimization Technique	***Lecture, conversation, exemplification***	**2**
7. Ants Colonies. Other natural paradigm	***Lecture, conversation, exemplification***	**2**
8. Application of evolutionary algorithms in optimization	***Lecture, conversation, exemplification***	**4**
9. Introduction to fuzzy logic. Fuzzy systems.	***Lecture, conversation, exemplification***	**4**
10. Introduction in Neural networks	***Lecture, conversation, exemplification***	**4**
Seminars-laboratories	**Teaching methods**	
1. Fundaments	***Project-work, computer-based activities, laboratory activities***	**4**
2. Genetic Algorithms. Description of the standard genetic algorithm. Fitness function. Specific operators: crossover, mutation, selection.	***laboratory activities***	**2**

3. Paradigm of Evolutionary Strategies. Specific operators.	***laboratory activities***	**2**
4. Genetic Programming. Examples	***laboratory activities***	**2**
5. Evolutionary Programming. Examples	***laboratory activities***	**2**
6. Artificial immune systems. Examples (Network Security).	***laboratory activities***	**2**
7. Particle Swarm Optimization technique. Examples	***laboratory activities***	**4**
8. Ant Colonies. Evolutionary approach for Travelling Salesman Problem. Other natural paradigm.	***laboratory activities***	**4**
9. Application of evolutionary algorithms in optimization. Multimodal Optimization, multi-criteria Optimization, Dynamic Optimization. - 2 laboratories	***laboratory activities***	**4**

References

1. Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., 1989.
2. Bäck T., Evolutionary Algorithms in Theory and Practice, Oxford University Press, 1996
3. Dumitrescu D., Lazzerini B., Jain L.C., Dumitrescu A., Evolutionary Computation, CRC Press, Boca Raton London, New York, Washington D.C., 2000
4. Rotar C., Modele naturale si algoritmi evolutivi, Ed. Accent, Cluj Napoca, 2008. (in Romanian, ppt presentation in English)

1. Corroboration of course contents with the expectations of the epistemic community's significant representatives, professional associations and employers in the field of the academic programme

- Currently there is a strong interest towards the development of intelligent software applications in various fields such as mobile phones, gaming industry, etc. Intelligent Computation discipline supports training of specialists in this direction, forming strategies and skills to apply intelligent algorithms where traditional methods are not effective.
- Coexistence of technical expertise within the University, particularly of specialization Applied Electronics is an additional reason to encourage the forming of the interdisciplinary and complementary teams.

2. Assessment

Activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Percentage of final grade
10.4 Course	*Final evaluation*	*Written paper*	60%
	-	-	-
10.5 Seminar/laboratory	*Continuous assessment*	*Laboratory activities portfolio*	40%
	-		-
10.6 Minimum performance standard: 5			
Implementation and documentation of the software units in high-level programming languages and efficiently used programming environments			

Submission date Course leader signature Seminar tutor signature

Date of approval by Department members Department director signature