COURSE PROGRAMME

1. Information about the programme

1.1 University	**University "Alexandru Ioan Cuza" of Iasi**
1.2 Faculty	**Faculty of Mathematics**
1.3 Department	**Department of Mathematics**
1.4 Domain	**Mathematics**
1.5 Cycle	**Masters**
1.6 Programme / Qualification	**Applied Mathematics**

2. Information about the course

2.1 Course Name			Evolutionary Computation and Genetic Algorithms				
2.2 Course taught by			Assoc. Prof. PhD. ANA MARIA MOSNEAGU				
2.3 Seminary / laboratory taught by			Assoc. Prof. PhD. ANA MARIA MOSNEAGU				
2.4 Year	I	**2.5** Semester	II	**2.6** Type of evaluation*	E	**2.7** Course type**	Op

E – Exam / C – Colloquium / V – Verification
***OB – Obligatory / OP – Optionally / F – Facultative*

3. Total hours (estimated per semester and activities)

3.1 Number of hours per week	4	**3.2** course	2	**3.3** seminary/ laboratory	2
3.4 Total number of hours	56	**3.5** course	28	**3.6** seminary/ laboratory	28

Distribution	hours
Individual study using textbooks, course notes, bibliography items, etc.	30
Supplimentary study (library, on-line platforms, etc.)	28
Individual study for seminary/laboratory, homeworks, projects, etc.	40
Tutoring	10
Examination	6
Other activities	5
3.7 Total hours of individual activity*	119
3.8 Total hours per semester	175
3.9 Credit points	7

4. Pre-requisites - Curriculum (if necessary)

Programming fundamentals
Algorithms and complexity
Data structures
Probabilities
Mathematical statistics

5. Conditions (if necessary)

5.1 Course	Lecture hall/classroom Laptop Video projector
5.2 Seminary / Laboratory	Computer laboratory (with MATLAB® and Python installed)

6. Objectives

Knowing, understanding and using evolutionary algorithms to solve concrete/real life problems Upon successful completion of this discipline, students will be able to: understand fundamental concepts of evolutionary computation/algorithms, describe evolutionary/genetic algorithms to solve concrete problems, implement these algorithms in MATLAB®/Python, test programs and interpret results

7. Specific competencies/Learning outcomes

• applies statistical analysis techniques • uses IT tools • uses application-specific interfaces • synthesizes information • manages interoperable, reusable, accessible, and easily findable data

8. Contents

8.1 Course	**Teaching methods**	**Remarks** (number oh hours, references)
Classifying computational problems: decision, search, optimization, modelling and simulation problems. P, NP, NP-complete and NP-hard problems	Exposition, conversation	2 hours
Evolutionary computation: the biological bases	Exposition, conversation	2 hours
Evolutionary algorithms: initialization, components (evaluation function, population models, parent selection mechanism, variation operators, survivor selection mechanism), termination condition) and classification	Exposition, conversation	2 hours
Binary representation and variation operators	Exposition, conversation	2 hours
Integer representation and variation operators	Exposition, conversation	2 hours
Floating-point representation and variation operators	Exposition, conversation	2 hours
Permutation representation and variation operators. Tree representation	Exposition, conversation	2 hours
Population models and selection operators	Exposition, conversation	2 hours
Evolutionary algorithm variants: a brief description, examples	Exposition, conversation	2 hours
Genetic algorithms: components, examples	Exposition, conversation	2 hours
Analyzing genetic algorithms and measuring their performance	Exposition, conversation	2 hours
Applications of genetic algorithms (optimization problems)	Exposition, conversation	2 hours
Applications of genetic algorithms (NP-complete problems)	Exposition, conversation	2 hours
Evolution strategies: components, self-adaptation mechanism, applications. Analyzing Evolution strategies and measuring their performance	Exposition, conversation	2 hours

Bibliography

C.-L. Cocianu, C.-R. Uscatu, Programare evolutivă și algoritmi genetici. Ediția a doua, revizuită și adăugită, Editura ASE, București, 2022
D.A. Coley, An Introduction to Genetic Algotithms for Scientists and Engineers, World Scientific, 1999
A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, 2nd Ed., Springer-Verlag, Heidelberg, 2015
D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading,1989
M. Mitchel, An Introduction to Genetic Algorithms, MIT Press, 1998

8.2 Seminary / Laboratory	**Teaching methods**	**Remarks** (number oh hours, references)

8.2 Seminary / Laboratory	Teaching methods	Remarks (number oh hours, references)
Intro to MATLAB®/Python	Exercise, conversation, programming	2 hours
Hill-climbing algorithm	Exercise, conversation, programming	2 hours
Simulated annealing algorithm	Exercise, conversation, programming	2 hours
Binary representation and variation operators	Exercise, conversation, programming	2 hours
Integer representation and variation operators	Exercise, conversation, programming	2 hours
Floating-point representation and variation operators	Exercise, conversation, programming	2 hours
Permutation representation and variation operators	Exercise, conversation, programming	2 hours
Population models and selection operators	Exercise, conversation, programming	2 hours
Evolutionary algorithm variants:examples	Exercise, conversation, programming	2 hours
Genetic algorithms: examples	Exercise, conversation, programming	2 hours
Analyzing genetic algorithms and measuring their performance	Exercise, conversation, programming	2 hours
Applications of genetic algorithms (optimization problems)	Exercise, conversation, programming	2 hours
Applications of genetic algorithms (NP-complete problems)	Exercise, conversation, programming	2 hours
Evolution strategies:examples	Exercise, conversation, programming	2 hours

Bibliography

C.-L. Cocianu, C.-R. Uscatu, Programare evolutivă și algoritmi genetici. Ediția a doua, revizuită și adăugită, Editura ASE, București, 2022
D.A. Coley, An Introduction to Genetic Algotithms for Scientists and Engineers, World Scientific, 1999
A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computing, 2nd Ed., Springer-Verlag, Heidelberg, 2015
D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading,1989
M. Mitchel, An Introduction to Genetic Algorithms, MIT Press, 1998

9. Coordination of the contents with the expectations of the community representatives, professional associations and relevant employers in the corresponding domain

The course syllabus is in accordance with the requirements formulated by the representatives of IT companies.

10. Assessment and examination

10.1 Continuous assessment	Percentage (min. 30%)	50

<table>
<tr><td rowspan="5">Course</td><td colspan="3">Assessment type</td><td>Mixed assessment</td></tr>
<tr><td colspan="3">Percentage</td><td>50</td></tr>
<tr><td colspan="3">Failure to pass the continuous assessment results in failure to pass the final assessment</td><td>No</td></tr>
<tr><td rowspan="5">Assessment methods</td><td>Details</td><td>Percentage</td><td>with reexamination</td></tr>
<tr><td>Project</td><td>30</td><td>No</td></tr>
<tr><td>Continuous practical assessment</td><td>30</td><td>No</td></tr>
<tr><td>Homework</td><td>30</td><td>No</td></tr>
<tr><td>Current assessment</td><td>10</td><td>No</td></tr>
<tr><td rowspan="5">Seminary / Laboratory</td><td colspan="3">Assessment type</td><td>Mixed assessment</td></tr>
<tr><td colspan="3">Percentage</td><td>50</td></tr>
<tr><td colspan="3">Failure to pass the continuous assessment results in failure to pass the final assessment</td><td>No</td></tr>
<tr><td rowspan="5">Assessment methods</td><td>Details</td><td>Percentage</td><td>with reexamination</td></tr>
<tr><td>Project</td><td>30</td><td>No</td></tr>
<tr><td>Continuous practical assessment</td><td>30</td><td>No</td></tr>
<tr><td>Homework</td><td>30</td><td>No</td></tr>
<tr><td>Current assessment</td><td>10</td><td>No</td></tr>
</table>

10.2 Final assessment	Percentage (max. 70%)	50
	Assessment type	Final written assessment

10.3 Special notes (special situations is assessment)	

10.4 Minimum performance standard	
1. Knowledge and understanding of the main concepts 2. Development of algorithms for solving concrete problems 3. Implement of developed algorithms using Python /MATLAB®	

Date, **Course coordinator,**
Assoc. Prof. PhD. ANA MARIA MOSNEAGU

Seminary coordinator,
Assoc. Prof. PhD. ANA MARIA MOSNEAGU

Aproval date in the department,

Head of the departament,
Prof. PhD. IONEL DUMITREL GHIBA