COURSE DESCRIPTION

1. Program Information

1.1 University	**Alexandru Ioan Cuza University of Iași**
1.2 Faculty	**Computer Science**
1.3 Department	**Computer Science**
1.4 Study Domain	**Computer Science**
1.5 Study Cycle	**Bachelor**
1.6 Study Program / Qualification	**Computer Science / Licentiate in Computer Science**

2. Course Information

2.1 Course Name			**Algorithm Design**				
2.2 Course Instructor			**Dorel Lucanu, Ștefan Ciobâcă**				
2.3 Tutorial Class Instructor			**Andrei Arusoaie**				
2.4 Study Year	**I**	**2.5** Semester	**2**	**2.6** Evaluation	**E**	**2.7** Course Status*	**OB**

** OB – Compulsory / OP – Optional*

3. Total estimated hours (hours per semester and didactic activities)

3.1 Hours per week	**4**	of which: **3.2** lecture	**2**	**3.3** tutorial/laboratory class	**2**
3.4 Hours in curriculum	**56**	of which: **3.5** lecture	**28**	**3.6**tutorial/laboratory class	**28**

Time Distribution	hours
Study of textbook, lecture notes, bibliography, and others	**14**
Supplementary documentation in the library, in electronic forums, and on the field	**14**
Preparation of tutorial/laboratories classes, homework, reports, portfolios and essays	**28**
Tutoring	-
Evaluation	**4**
Other activities	-

3.7 Total hours of individual study	**56**
3.8 Total hours per semester	**116**
3.9 Credits	**5**

4. Preconditions (if any)

4.1 Curriculum	**Data Structures**
4.2 Skills	**- the knowledge of the following data structures: lists, trees, graphs (as data structures), heaps, union-find and their associated operations**

5. Conditions (if any)

5.1 Course Operation	
5.2 Tutorial/Laboratory Class Operation	**Compulsory attendance at tutorial classes**

6. Specific Skills Acquired

Professional Skills	**C1.** The ability to design algorithms for common computational problems. **C1.** The ability to use mathematical methods for analysing an algorithm. **C3.** The ability to adapt standard algorithms to specific problems. **C4.** The ability to evaluate the complexity of a computational problem.

Transversal Skills	CT1. The ability to design algorithms for problems whose domain belongs to other disciplines (eg computational geometry, mathematical programming, word processing). CT2. The ability to use mathematical tools (mathematical analysis, algebra, probabilities, logic) for algorithm analysis.

7. Course Objectives (from the grid of specific skills acquired)

7.1 General Objectives	Developing algorithmic thinking and of basic techniques for designing and analyzing algorithms.
7.2 Specific Objectives	O1. The knowledge of the concept of computational model and of the main functions for measuring the efficiency of algorithms. O2. The knowledge of the main types of algorithms (deterministic, non-deterministic, probabilistic). O3. The knowledge of the main algorithms design paradigms. O4. The knowledge of the concept of complexity of a problem and the classification of problems depending on their complexity.

8. General Description

8.1	**Course**	**Teaching Methods**	**Observations** (hours and bibliographic references)
1.	Executable Algorithmic Language	Lecture	2
2.	Computational Problem Solved by an Algorithm. Complexity of Algorithms. Worst-case complexity	Lecture	2
3.	Computational Complexity of Problems. Problem Reduction	Lecture	2
4.	Nondeterministic Algorithms. Probabilistic Algorithms	Lecture	2
5.	Average-Case Complexity	Lecture	2
6.	Domain Specific Algorithms: Strings (I)	Lecture	2
7.	Domain Specific Algorithms: Strings (II)	Lecture	2
8.	Midterm Examination	Written test	2
9.	Greedy Algorithms	Lecture	2
10.	Dynamic Programming (I)	Lecture	2
11.	Dynamic Programming (II)	Lecture	2
12.	NP-complete Problems	Lecture	2
13.	Algorithms Computing Exact Solutions of NP-complete Problems. Backtracking and Branch-and-bound	Lecture	2
14.	Approximation Algorithms	Lecture	2

Bibliography

Main references:

Dorel Lucanu, Mitica Craus. Proiectarea algoritmilor. Polirom, 2008.
T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introduction to Algorithms, MIT Press, 1990.
T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introducere in Algoritmi,Computer Libris Agora, 2000.
M. CROCHEMORE, C. HANCART, T. LECROQ: Algorithms on Strings, Cambridge University Press

Supplementary references:

1. Manuel Blum, Robert W. Floyd, Vaughan Pratt, Ronald L. Rivest, and Robert E. Tarjan. 1972. Linear time bounds for median computations.
2. Manuel Blum, Robert W. Floyd, Vaughan Pratt, Ronald L. Rivest, and Robert E. Tarjan.Time Bounds for Selection.
3. CMU 15-451 (Algorithms), Fall 2011. Lecture 4. Selection (deterministic & randomized): finding the median in linear time.

8.2	**Tutorial / Laboratory Class**	**Teaching methods**	**Observations** (hours and bibliographic references)
1.	Examples of elementary algorithms	Review of the topics presented at the lecture, proposing a set of exercises, individual work, interactive methods on the board	2
2.	Examples of problems solved by an algorithm. Calculating the complexity of the discussed algorithms.	Idem	2
3.	Computational Complexity of the Problems. Examples of Problem Reduction	Idem	2
4.	Nondeterministic Algorithms. Probabilistic Algorithms	Idem	2
5.	Computing the Average Case Complexity	Idem	2
6.	Discussing Domain Specific Algorithms: Strings (I)	Idem	2
7.	Discussing Domain Specific Algorithms: Strings (II)	Idem	2
8.	Midterm Examination	Discussion on the solutions of the problems from the written test	2
9.	Examples of Greedy Algorithms	Review of topics presented in the course, Propose a set of exercises, Individual work, Interactive methods on the board	2

10.	Examples of Dynamic Programming Algorithms (I)	Idem	2
11.	Examples of Dynamic Programming Algorithms (I)	Idem	2
12.	Examples of NP-complete Problems	Idem	2
13.	Examples of Algorithms Computing Exact Solutions of NP-complete Problems	Idem	2
14.	Examples of Approximation Algo-rithms	Idem	2

Bibliography

Dorel Lucanu, Mitica Craus. Proiectarea algoritmilor. Polirom, 2008.
T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introduction to Algorithms, MIT Press, 1990.
T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introducere in Algoritmi,Computer Libris Agora, 2000.
M. CROCHEMORE, C. HANCART, T. LECROQ: Algorithms on Strings, Cambridge University Press

9. Course content synchronization with the expectations of the community representatives, professional associations and employers from the program domain

The content of the discipline is in accordance with the topics of hiring interviews in IT companies.

10. Evaluation

Activity Type	**10.1 Evaluation criteria**	**10.2 Evaluation methods**	**10.3** The **weight of each evaluation form (%)**

10.4 Course	- Correct understanding of the concepts of computational model, algorithm, problem solved by an algorithm, complexity - Ability to compare two problems from an algorithmic point of view - Correct understanding of the problems and algorithms for them (presented at the lecture) - Quality of the answers	Written tests	80 %
10.5 Tutorial/ Laboratory Class	- Ability to write an algorithm - Ability to analyze an algorithm - Ability to apply paradigms for designing new algorithms - Algorithms description quality	Presence Assessment of classroom activity Bonus (max 10%)	20 %
10.6 Minimal performance standards			
The following criteria must be met simultaneously for graduating: • At least 35 points as the sum of test scores (minimum 35 out of 80) • At least 10 points for seminar activity (minimum 10 out of 30) Final marks are set according to ECTS criteria.			

Date	Course Instructor	Tutorial/Laboratory Instructor
15 February 2018	Prof.univ.dr. Dorel Lucanu Conf. Dr. Ștefan Ciobâcă	Lect. Dr. Andrei Arusoaie

Department Date of Approval Director of the Department