SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Masters
1.6 Program of study / Qualification	Data Science / Masters
1.7 Form of education	Full time

2. Data about the subject

2.1 Subject name	***Algorithms***			Subject code	**2**
2.2 Course responsible / lecturer	Prof. dr. eng. Potolea Rodica - Rodica.Potolea@cs.utcluj.ro				
2.3 Teachers in charge of seminars / Laboratory / project	Assoc. prof. dr. eng. Oprișa Ciprian - Ciprian.Oprisa@cs.utcluj.ro				
2.4 Year of study	I	2.5 Semester	1	2.6 Type of assessment (E - exam, C - colloquium, V – verification)	E
2.7 Subject category	Formative category: DF – fundamental, DS – speciality, DC – complementary				DS
	Optionality: DOB – mandatory, DOP – optional (alternative), DFA – optional (free choice)				DOB

3. Estimated total time

3.1 Number of hours per week	3	of which:	Course	1	Seminars	-	Laboratory	1	Project	1
3.2 Number of hours per semester	42	of which:	Course	14	Seminars	-	Laboratory	14	Project	14
3.3 Individual study:										
(a) Manual, lecture material and notes, bibliography										20
(b) Supplementary study in the library, online and in the field										9
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays										20
(d) Tutoring										4
(e) Exams and tests										5
(f) Other activities:										-
3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))										58
3.5 Total hours per semester (3.2+3.4)										100
3.6 Number of credit points										4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Data Structures, Programming Techniques, Algorithms
4.2 Competence	Algorithms performance evaluation, Knowledge of fundamental algorithms on basic data structures

5. Requirements (where appropriate)

5.1. For the course	Blackboard, projector, computer
5.2. For the applications	Computers

6. Specific competence

6.1 Professional competences	analyse big data • analyse business processes • analyse decentralised applications • build predictive models • create data models • debug software • define software architecture • define technical requirements • design cloud architecture • design process for blockchain-based systems • develop blockchain innovative architectures • develop software prototype • develop with cloud services • evaluate blockchain architectures • explain blockchain implications • explain distributed ledger technologies principles • identify blockchain innovation opportunities • implement smart contracts • interpret technical requirements • manage cloud data and storage • outline blockchain-based identity management • oversee development of software • perform data cleansing • perform data mining • perform scientific research • provide technical documentation • recognise blockchain application areas • recognise blockchain risks • use data processing techniques • use software design patterns • use software libraries • utilise computer-aided software engineering tools • utilise machine learning
6.2 Cross competences	The graduate: • develop an analytical approach • taking a proactive approach • developing strategies to solve problems • being open minded • coordinate engineering teams

7. Expected Learning Outcomes

Knowledge	The student has knowledge of: • algorithms • task algorithmisation • data mining • digital data processing
Skills	The student is able to: • create data sets • manage databases • create data models • develop statistical software

Responsibilities and autonomy	The student has the ability to work independently in order to: • develop an analytical approach • take a proactive approach • develop strategies to solve problems • be open minded • coordinate engineering teams

8. Discipline objective (as results from the *key competences gained*)

8.1 General objective	Acquiring problem-solving skills specific to data analysis tasks, identifying potential solutions and selecting appropriate ones.
8.2 Specific objectives	• Complexity computation • Distinguishing between complexity classes • Approximating computationally hard problems • Knowledge of algorithms specific to big data

9. Contents

9.1 Lectures	Hours	Teaching methods	Notes
Algorithms complexity	2		
Complexity classes; P vs. NP, NP-Completeness	2		
NP-complete problems; reduction algorithms	2		
NP-complete problems approximation	2		
Sublinear algorithms for big data	2		
Linear programming	2		
Searching in big data	2		

Bibliography:

1. Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.
2. Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive data sets*. Cambridge university press, 2020.
3. Feldman, Moran. *Algorithms for Big Data*. World Scientific, 2020.

9.2 Applications - Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Experimental analysis of the algorithms complexity	2		
Identifying the algorithms complexity classes	2		
Reduction algorithms for NP-Complete problems	2		
Approximation algorithms for NP-Complete problems	2		
Implementation of sublinear algorithms	2		
Implementation and analysis of linear programming techniques	2		
Algorithms for searching in big data	2		
Introduction and presentation of the project theme	2		
Bibliographic research: resources identification	2		
Bibliographic research: documentation analysis and synthesis	2		
Requirements gathering	2		
Implementing and testing a case study	2		
Analysis of the experimental results	2		
Redacting and presenting the final document	2		

Bibliography:

1. Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.
2. Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive data sets*. Cambridge university press, 2020.
3. Feldman, Moran. *Algorithms for Big Data*. World Scientific, 2020.

**Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.*

10. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

11. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Theoretical analysis and problem solving skills based on the acquired knowledge	Written exam	50%
Seminar	-	-	-
Laboratory	Based on the obtained results and the analysis and synthesis skills	Practical problem-solving skills	20%
Project	Based on the obtained results and the analysis and synthesis skills	Oral evaluation	30%

Minimum standard of performance:

- Laboratory ≥ 5
- Project evaluation ≥ 5
- Final exam ≥ 5;
- Final grade computation: 50% final exam + 20% laboratory + 30% project

Date of filling in: 29.04.2026	**Responsible**	**Title First name Last name**	**Signature**
	Course	Prof.dr.eng. Rodica POTOLEA	
	Applications	Assoc.prof.dr.eng. Ciprian OPRIȘA	

Date of approval in the department	Head of department, Prof.dr.eng. Rodica Potolea
Date of approval in the Faculty Council	Dean, Prof.dr.eng. Vlad Mureșan