

SYLLABUS

- 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	Automation
1.4	Field of study	Systems Engineering
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Automation and Applied Informatics
1.7	Form of education	Full time
1.8	Subject code	19.00

- Data about the subject**

2.1	Subject name		Algorithm Design					
2.2	Subject area		Algorithm Design					
2.3	Course responsible/lecturer		Sl.dr.eng. Cornelia Melenti, cornelia.melenti@cs.utcluj.ro					
2.4	Teachers in charge of seminars		Sl.dr.eng. Cornelia Melenti, cornelia.melenti@cs.utcluj.ro					
2.5	Year of study	2	2.6 Semester	1	2.7 Assessment	E	2.8 Subject category	DF/DOB

- Estimated total time**

3.1	Number of hours per week	4	3.2 of which, course:	2	3.3 applications:	2
3.4	Total hours in the curriculum	56	3.5 of which, course:	28	3.6 applications:	28
Individual study				hours		
Manual, lecture material and notes, bibliography				28		
Supplementary study in the library, online and in the field				14		
Preparation for seminars/laboratory works, homework, reports, portfolios, essays				14		
Tutoring				14		
Exams and tests				4		
Other activities						
3.7	Total hours of individual study		74			
3.8	Total hours per semester		130			
3.9	Number of credit points		5			

- Pre-requisites (where appropriate)**

4.1	Curriculum	Computer programming
4.2	Competence	Knowledge of programming in C

- Requirements (where appropriate)**

5.1	For the course	Video projector, whiteboard.
-----	----------------	------------------------------

		Attendance at the course is mandatory.
5.2	For the applications	Computers installed with Code Blocks. White or blackboard Attendance at the laboratory is mandatory.

• **Specific competences**

Professional competences	C2 Working with fundamental concepts from computer science, information and communication technology. C2.1 Description of the operation and structure of computer systems, communication networks and their applications in systems engineering using knowledge of programming languages, environments and technologies, programming engineering and specific tools (algorithms, schemes, models, protocols, etc.).
Cross competences	N/A

• **Discipline objectives (as results from the *key competences gained*)**

7.1	General objective	<ul style="list-style-type: none"> The study abstract data structures and specific algorithms
7.2	Specific objectives	<ul style="list-style-type: none"> Acquisition of theoretical knowledge on: <ul style="list-style-type: none"> an elementary data types and their representation in memory an abstract data structures: sets, lists, stacks, queues priority, graphs, trees, hash tables operations on an abstract textual data: insertions, deletions, searches, updates, traversal (depth, width), minimum cost problems in graphs, sorting algorithms a programming techniques: divide et impera and recursion, backtracking, greedy, etc. an evaluation algorithms, optimal scheduling problems Acquisition of practical skills in: <ul style="list-style-type: none"> creating an algorithmic thinking analysis of a problem finding solutions to a problem that its implementation in a C language and testing the chosen solution

• **Contents**

8.1. Lecture (syllabus)			Teaching methods	Notes
•	Introduction. Analysis and formalization of problems. Algorithms: definitions, basics, performance.	2	Course presentations based on slides using the video projector. Drawing and explaining algorithms with examples, on the board.	
•	Lists. Types and Representations: Singly and Doubly Linked Dynamic Lists. Scrolling through lists.	2		
•	Operations on lists (account): insert, delete, update item. Stacks and Queues: Specific Operations.	2		
•	Fundamental sorting algorithms: bubble sort, insertion sort, sorting by selection, merge sort, quicksort, counting sort, radix sort	2		
•	Sets. Operations on sets	2		
•	Trees. Types of trees: binary trees, perfect balance tree, AVL trees. Create and trees representation in memory. Trees traversing. Operating on trees: insertion, deletion, update	2		
•	Search algorithms in a tree. Applications of trees in linguistics: 2-3 and 2-3-4 trees. Applications in coding: Coding Huffman	2		
•	Graphs: types and representation. Traversing the graph. Operations on graphs: insert, delete, update	2		
•	Problem solving using graphs. Minimum road (Dijkstra, Floyd algorithmis), minimum spanning tree (Kruskal, Prim algorithms)	2		
•	Hash table. Representation and aplication	2		
•	General methods of developing algorithms: Recursive algorithms, Backtracking method.	2		
•	General methods for developing algorithms: The divide et impera method.	2		
•	General methods of developing algorithms: The greedy method.	2		
•	General methods of developing algorithms: Branch and bound method.	2		
Bibliography				
<ul style="list-style-type: none"> • IGNAT Iosif, IGNAT Claudia-Lavinia - Structuri de date si algoritmi, 2007 Cota 527.366 • IGNAT Iosif, IGNAT Claudia-Lavinia - Structuri de date si algoritmi : indrumator de lucrari de laborator, 2001 Cota 506.016 				

	<ul style="list-style-type: none"> • JOLDOS Marius, IGNAT Iosif - Data structures and algorithms : laboratory guide, 2003 Cota 509.111 • IGNAT Iosif, IGNAT Claudia-Lavinia - Programarea calculatoarelor : descrierea algoritmilor si fundamentele limbajului C/C++ , Cota 508.311 • N. Wirth, “Algorithms and Data Structures”, http://www.ethoberon.ethz.ch/WirthPubl/AD.pdf • CRAUS Mitica, BARSAN Corneliu - Structuri de date si algoritmi, 2002 Cota 507.305 • CORMEN Thomas H., LEISERSON Charles E. , RIVEST Ronald R. - Introducere in algoritmi, 2000 Cota 501.507 • KNUTH Donald E. - Arta programarii calculatoarelor. Vol. 1: Algoritmi fundamentali, 2000 Cota 501.199/1 • KNUTH Donald E. - Arta programarii calculatoarelor. Vol. 3: Sortare si cautare, 2000 			
8.2. Applications/Seminars			Teaching methods	Notes
•	Control structures. Using the menus in programming. Vectors and matrix. Working with files (C review)	2	Assessing knowledge from previous laboratory (tests) and mapping (where applicable) of specific algorithms (15 min) Implement algorithms in current laboratories	C language is used for implementation (CodeBlocks) (MS Teams and Whiteboard fox for online)
•	String operation. Dynamic allocation of memory (C review).	2		
•	Single linked lists	2		
•	Double linked lists (Test 1)	2		
•	Sorting algorithms	2		
•	Sorting algorithms (Test 2)	2		
•	Tree representation and traversals. Evaluation of expression	2		
•	Binary tree (Test 3)	2		
•	Graph representation and traversals	2		
•	Directed and Undirected graphs algorithms (Test 4)	2		
•	Hash Table	2		
•	Algorithms development methods	2		
•	Algorithms development methods	2		
•	Final test (Colocvium)	2		
•	Bibliography IGNAT Iosif, IGNAT Claudia-Lavinia - Structuri de date si algoritmi : indrumator de lucrari de laborator, 2001 Cota 506.016 JOLDOS Marius, IGNAT Iosif - Data structures and algorithms : laboratory guide, 2003 Cota 509.111			

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

• **Evaluation**

Activity type	10.1 Assessment criteria	10.2 Assessment methods	10.3 Weight in the final grade
Course	The degree of knowledge of the concepts presented in the course How they are applied in solving practical problems	Writing exam: Theory 50% 1 problem 50%	70%
Applications	Acquiring specific knowledge and ability to implement in practice. Correct implementation of algorithms	4 tests 50% Final test 50%	30%
10.4 Minimum standard of performance			
Minimum 5 for exam and lab OBS: students can receive bonuses based on participation and obtaining special results in recognized competitions in the field of Automation, Computers or Mathematics			

Date of filling in
15.06.2024

Teachers in charge of seminars
PhD eng. S.L. Cornelia Melenti

Date of approval in the department Automation

Head of department
PhD eng. Prof. Honoriu Valean

Date of approval in the faculty
Automation and Computer Science

Dean of the Faculty of Automation and
Computer Science
Prof.dr.ing. Mihaela DINSOREANU
