NAME OF THE COURSE	**Data Structures and Algorithms**					
Code	PMIE10	Year of study				
Course teacher	prof.dr. sc. Marko Rosić	Credits (ECTS)	6,0			
Associate teachers		Type of instruction (number of hours)	L	S	E	F
			30		30	
Status of the course		Percentage of application of e-learning				

COURSE DESCRIPTION	
Course objectives	Understand, acquire and learn algorithm and data structure concepts. Understand and learn application and implementation of the algorithms and data structures.
Course enrolment requirements and entry competences required for the course	Course: Programming I Competencies: Basic object-oriented concepts, C# programming language
Learning outcomes expected at the level of the course (4 to 10 learning outcomes)	1. Classify data structures 2. Classify basic algorithms 3. Define data structures 4. Apply algorithms and data structures 5. Learn how to extend existing data structures
Course content broken down in detail by weekly class schedule (syllabus)	Week 01 Labs: Pre-test, course overview Lectures: Algorithms, algorithm complexity Week 02: Labs: memory allocation, stack and queue (add and remove elements) Lectures: Sorting algorithms Week 03: Labs: Sorting algorithms implementation Lectures: data structures overview, linear data structures, non-linear data structures, collections, trees, graphs Week 04: Labs: using built in queue and stack classes, linked lists Lectures: using C# ArrayList, Stack, Queue, hashtable Week 05: Labs: extending existing data structure classes (for example: add method AddSorted into LinkedList) Lectures: Dictionary, SortedList, design hastable class Week 06: Dictionary, SortedList, design hastable class Labs: Lectures: binary trees implementation, binary trees algorithms Week 07: Labs: input data into binary search tree Lectures: delete nodes, rotations Week 08: Labs: midterm exam Lectures: balancing trees(AVL, CC) Week 09: Labs: priority queue, heap, heap sort Lectures: heap (recursive and non recursive), priority queue Week 10: Labs: binary trees, tree height, rotation, draw tree

	Lectures: trees with multiple child nodes, graphs Week 11: Labs: implement graph data structure, graph traversals, depth first search, breadth first search Lectures: different graph implementation (adjacency matrix and linked lists, minimum spanning tree Week 12: Labs: shortest path algorithm, greedy algorithm Lectures: graph types, shortest path algorithm Week 13: Labs: breadth first search (search for friends, Bacon number) Lectures: knapsack problem Week 14: Labs: practice for final exam Lectures: backtracking, dynamic programming Week 15: Labs: final exam Lectures: preparation for the final exam	
Format of instruction	☒ lectures ☐ seminars and workshops ☒ exercises ☐ on line in entirety ☐ partial e-learning ☐ field work	☐ independent assignments ☐ multimedia ☐ laboratory ☐ work with mentor ☐ homework assignments
Student responsibilities	Attendance, active participation in learning and teaching process, midterm exams, practical exam, oral exam	

Screening student work *(name the proportion of ECTS credits for each activity so that the total number of ECTS credits is equal to the ECTS value of the course)*

Name	Ects	Name	Ects	Name	Ects
Class attendance	1	Research		Experimental work	
Oral exam	2	Report		Homework assignments	
Seminar essay		Essay			
Tests		Practical training			
Written exam	2	Project			

Grading and evaluating student work in class and at the final exam	Practical exam consists of two parts, first (midterm exam) represents 40% of the final practical exam grade, and second part represents 60% of the practical exam grade. Students that fail at one or both parts of the practical exam during semester only write part which they did not pass. Everyone must also pass the oral exam which is 20% of the final grade.

Required literature (available in the library and via other media)

Title	Number of copies in the library	Availability via other media
	0	

Optional literature (at the time of submission of study programme proposal)	Robert Manger: Strukture podataka i algoritmi (dostupno online), M. McMillan: Data Structures and Algorithms Using C#, 2007 Nastavni materijali dostupni na Internetu.
Quality assurance methods that ensure the acquisition of exit competences	Talk with students, student evaluation using the anonymous survey, the success of students in the exam, self-assessment.
Other (as the proposer wishes to add)	