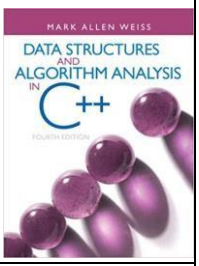
Identification	**Department**	Computer Science
	Program	Graduate
	Subject	**Advanced Data Structure,** 4 KU (8 ECTS) credits
	Term	Fall 2019
	Instructor	**PhD, Associate Professor Leyla Muradkhanli**
	Classroom/hours	
Prerequisites	Programming Languages	
Language	English	
Compulsory/Elective	Major	
Text books and course materials	**Textbook** Data Structures and Algorithm Analysis in C++ by Mark Allen Weiss, 4th edition, Pearson, 2014. 	

Teaching methods	**Case analysis**	
	Group discussion	**x**
	Lab	**x**
	Lecture	**x**
	Course paper	**x**
	Others	

Evaluation Criteria	**Methods**	**Date/deadlines**	**Percentage (%)**
	Midterm Exam		**30%**
	Case studies		
	Class Participation		
	Quizzes		**10%**
	Project		
	Presentation		
	Laboratory Work (Assignments)		**20%**
	Final Exam		**40%**
	Other		
	Total		**100%**

Course objectives

This is an advanced course in data structures. The main objective of this course is to provide students with the background to fundamental design, analysis, and implementation of data structures and algorithms.

This course specifically has the following objectives:

- The analysis and evaluation of the data structure needs of particular problems;
- The design, analysis, and implementation of C++ programs by using data structures and algorithms.

Learning outcomes	Upon successfully completed this course, the student will be able to: • Apply advance C ++ programming techniques to developing solutions for particular problems; • Design and implement abstract data types such as linked list, stack, queue and tree by using C++ as the programming language using static or dynamic implementations; • Analyse, evaluate and choose appropriate abstract data types and algorithms to solve particular problems; • Design and implement C++ programs that apply abstract data types.

Course outline	Abstract Data Types. List. Stack. Queue. Trees. Hashing. Priority Queues (Heaps). Sorting. Graph Algorithms. Algorithm Design Techniques. Advanced Data Structures and Implementation.

Tentative Schedule

Week	Date	Topics	Textbook/Assignments
1	21.09.19	**Course Overview and Introduction.**	Presentation Chapter 1
2	28.09.19	**Algorithm Analysis** Abstract Data Types (ADTs)	Chapter 2
3	05.10.19	**The List ADT** Simple Array Implementation of Lists Simple Linked Lists vector and list in the STL Implementation of vector Implementation of list	Chapter 3
4	12.10.19	**The Stack ADT** Stack Model Implementation of Stacks Applications	Chapter 3
5	19.10.19	**The Queue ADT** Queue Model Array Implementation of Queues Applications of Queues	Chapter 3
6	26.10.19	**Trees** Implementation of Trees Tree Traversals with an Application Binary Trees Binary Search Trees	Chapter 4

7	02.11.19	**Associative Containers** set and multiset map and multimap Implementation of set and map	Chapter 4
8	09.11.19	**Midterm exam**	
9	16.11.19	**Hashing** General Idea Hash Function Separate Chaining Hash Tables without Linked Lists Rehashing Hash Tables in the Standard Library	Chapter 5
10	23.11.19	**Priority Queues (Heaps)** Simple Implementations Binary Heap Applications of Priority Queues	Chapter 6
11	30.11.19	**Sorting. Elementary Sorting Algorithms.** Insertion Sort Selection Sort Bubble Sort	Chapter 7
12	07.12.19	**Sorting. Efficient Sorting Algorithms.** Shell Sort Heap Sort Mergesort Quicksort	Chapter 7
13	14.12.19	**Graph Algorithms** Graph Representation Graph Traversals Shortest-Path Algorithms Spanning Trees Connectivity Topological Sort Network Flow Problems	Chapter 9
14	21.12.19	**Algorithm Design Techniques** Greedy Algorithms Divide and Conquer Dynamic Programming Randomized Algorithms Backtracking Algorithms	Chapter 10

15	28.12.19	**Advanced Data Structures and Implementation** Top-Down Splay Trees Red-Black Trees	Chapter 12
		Final exam	