	Faculty of Engineering and Natural Sciences Computer Engineering Department	Syllabus

COURSE INFORMATION

Academic Year / Semester		2022 – 2023 / Fall
Course Code, Name, Credits (T-A-C)		1413211001 Data Structures (3-0-3)
Lecture and Lab Schedules	**Theoretical**	
	Lab	
Course Type		Compulsory
Prerequisite		Programming II (OOP)

ACADEMIC PERSONAL

Title / First and Last Name	Prof. Dr. Haluk Gümüşkaya
Room	A7-12
E-Mail Address / Web Page	**E-mail:** haluk.gumuskaya@atlas.edu.tr and haluk@gumuskaya.com **Course Web Page:** Google Classroom (TBA)
Telephone	0850 450 3439-1316 / 444 34 39 - 1316
Office Hours	TBA
Assistant	-

COURSE DESCRIPTION

Abstract data types, arrays, linked lists, stacks and queues, tree structures, introduction to graphs. Recursive applications. Study of the basic data structures and their implementations. Classification of data structures, memory and time considerations. Basic searching and sorting and algorithms.

COURSE OBJECTIVES

Upon successful completion of the course, the student is expected to:

- How to measure the algorithm time/space complexity using the Big O, Ω, and Θ notations.
- Has understand the concept of divide and conquer wrt algorithms.
- Learn and use classic data structures in applications: array lists, linked lists, stacks, queues, priority queues, sets, maps, binary trees, and hashing.
- Understand the concept of a hash function and its application in search.
- Understand the concept of a tree, a forest, a singly rooted tree, and a binary tree
- Understand the concept of ancestral relationships in trees
- Understand the concept of a binary search tree, and modify a binary tree to a binary search tree
- Has ability to understand, and to implement linear and binary searches
- Ability to program preorder, inorder, and postorder binary tree traversals
- Has ability to understand, and to implement basic sorting and selection algorithms
- Understand concepts in graphs: directed/ undirected/ oriented, graphs and chains/ paths/ cycles.
- Understand the concepts of weakly and strongly connected graphs.
- Understand Dijkstra's shortest path algorithm
- Ability to understand the concept of a minimal spanning tree and to find a minimal spanning trees for a graph

	Faculty of Engineering and Natural Sciences Computer Engineering Department	Syllabus

COURSE TEXTBOOK AND OTHER MATERIALS

Main Textbook	1. *Data Structures and Algorithms in Python*, 1th Ed., M. T. Goodrich, R. Tamassia, M. H. Goldwasser, Wiley, 2013. https://www.wiley.com/en-us/Data+Structures+and+Algorithms+in+Python-p-9781118549582
Lab Resources	PyCharm 2022.1.2 (Professional Edition) (Python IDE), Eclipse (Java IDE)
Recommended Books and other Resources	2. *Hands on Data Structures and Algorithms with Python*, 3rd Ed. 2022. 3. *Algorithms*, 4th Ed, R. Sedgewick, Addison-Wesley, 2011. https://algs4.cs.princeton.edu/lectures/ https://cuvids.io/app/video/97/watch/ Sedgewick's Coursera free courses based his book: Algorithms Part I and Algorithms Part II https://www.coursera.org/learn/algorithms-part1 https://www.coursera.org/learn/algorithms-part2 4. *Introduction to Algorithms*, 4th Ed., The MIT Press (2022) Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 2022. 5. *Data Structures and Algorithms Specialization* (6 courses), Coursera https://www.coursera.org/specializations/data-structures-algorithms 6. *Data Structures and Algorithms in Java*, 6th Ed., M. T. Goodrich, R. Tamassia, M. H. Goldwasser, Wiley, 2014. 7. *Data Structures and Problem Solving using Java (4th Ed.),* Mark A. Weiss, Addison-Wesley, 2014.

WEEKLY LECTURE SCHEDULE (Tentative)

W	D	Topics Covered	Activities
1	4-Oct	**01.1-Course Overview** **01.2- Introduction: Basic Programming Model** **01.3- Introduction: Programming Model: OOP**	**Getting Started:** **Learning Python and its IDE: PyCharm** **Introduction to Python and PyCharm** Python and PyCharm Installations and Tutorials Finish first 2 chapters of the textbook
2	11-Oct	**02. Algorithm Analysis and Recursion** **02.1. Algorithm Analysis** Experimental Studies, The Seven Functions Used in the Course, Asymptotic Analysis, Simple Justification Techniques **02.2. Recursion** Illustrative Examples, Analyzing Recursive Algorithms, Recursive Run Amok, Examples of Recursion: Linear, Binary ve Multiple Recursions, Designing Recursive Algorithms	

3	**18-Oct**	**03-Array Based Sequences** Python's Sequence Types, Low-Level Arrays, Dynamic Arrays and Amortization, Efficiency of Python's Sequence Types, Using Array Based Sequences, Multidimensional Data Sets	
4	**25-Oct**	**04-Stacks, Queues and Deques** Stacks, Queues and Double-Ended Queues	
5	**1-Nov**	**05-Linked Lists** Single Linked Lists, Circularly Linked Lists, Double Linked Lists, The Positional List ADT, Sorting a Positional List, Linked-Based vs. Array-Based Sequences	
6	**8-Nov**	**06-Trees (1-2)** General Trees, Binary Trees, Implementing Trees,	
7	**15-Nov**	**07-Trees (2-2)** Tree Traversal Algorithms	
8	**22-Nov**	**Midterm Exam**	
9	**29-Nov**	**09-Priority Queues** The Priority Queue Abstract Data Type, Implementing a Priority Queue, Heaps, Sorting with a Priority Queue, Adaptive Priority Queue	
10	**6-Dec**	**10-Maps, Hash Tables and Skip Lists** Maps and Dictionaries, Hash Tables, Sorted Maps, Skip Lists, Sets, Multisets, and Multimaps	
11	**13-Dec**	**11-Search Trees** Binary Search Trees, Balanced Search Trees, AVL Trees, Splay Trees, (2, 4) Trees, Red-Black Trees,	
12	**20-Dec**	**12-Sorting and Selection Algorithms** Why Study Sorting Algorithms, Merge-Sort, Quick-Sort, Studying Sorting though an Algorithmic Lens, Comparing Sorting Algorithms, Python's Built-In Sorting Functions	

13	**27-Dec**	**13-Graph Algorithms (1-2)** Graphs, Data Structures for Graphs, Graph Traversals: DFS, DFS Implementation and Extensions, BFS, Transitive Closure, Directed Acyclic Graphs,	
14	**3-Jan**	**14-Graph Algorithms (2-2)** Shortest Paths, Minimum Spanning Trees.	
15	**10-Jan**	**Project Demonstrations / Midterm Exam II**	
	16-27 Jan	**Final Exam**	

WEEKLY LABORATORY SCHEDULE

W	D	Lab	Lab Experiments

GRADING

Practical Part:

- %20: Weekly Flipped Classroom Activities
- %30: Project

Theoretical Part:

- %20: Midterm Exam
- %30: Final Exam