	Faculty of Engineering and Natural Sciences Computer Engineering Department	Syllabus

COURSE INFORMATION

Academic Year / Semester		2022 – 2023 / Spring
Course Code, Name, Credits (T-A-C)		1413211001 Design and Analysis of Algorithms (3-0-3)
Lecture and Lab Schedules	**Theoretical**	
	Lab	
Course Type		Compulsory
Prerequisite		Data Structures

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

Introduction: Historical Algorithms, Algorithm Analysis, Greedy Algorithms, Divide-and-Conquer, Dynamic Programming, Text Processing, Graph Algorithms: Graphs and Traversals, Directed Graphs, Shortest Paths, Minimum Spanning Trees, Graph Algorithms in AI Search Problems-Maze Traversal, Graph Applications, Constraint Satisfaction Problem.

COURSE OBJECTIVES

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

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

WEEKLY LECTURE SCHEDULE (Tentative)

W	D	Topics Covered	Activities
1		**01.1-Course Overview** **01.2-Historical Algorithms and Algorithm Analysis** Greatest Common Divisors, Fibonacci Numbers, Computing Runtimes: Algorithm Analysis, Algorithm Design is Hard	
2		**02-Greedy Algorithms** Introduction to Gready Algorithms, Finding the Largest Number, Car Fueling, Design and Analysis of the Greedy Algorithm, Main Ingredients, Grouping Children, Fractional Knapsack, Summary	
3		**03-Divide-and-Conquer-Searching and Polynomial Multiplication** **03.1-Divide-and-Conquer-Searching in an Array** Main Idea of Divide-and-Conquer, Linear Search in Array, Binary Search in a Sorted Array **03.2-Divide-and-Conquer-Polynomial Multiplication** Problem Overview, Naive Algorithm, Naive Divide-and-Conquer Algorithm, Faster Divide-and-Conquer	
4		**04-Divide-and-Conquer-Master Theorem and Sorting** **04.1-Divide-and-Conquer-Master Theorem** What is the Master Theorem? Proof of Master Theorem **04.2-Divide-and-Conquer-Sorting** Problem Overview, Selection Sort, Merge Sort, Lower Bound for Comparison Based Sorting, Non-Comparison Based Sorting Algorithm **04.3-Divide-and-Conquer-Quick Sort** Overview, Algorithm, Random Pivot, Running Time Analysis, Equal Elements, Final Remarks	

5		**05-Dynamic Programming** **05.1-Dynamic Programming-Change Problem.pptx** Greedy Change, Recursive Change, Dynamic Programming **05.2-Dynamic Programming-String Comparison** The Allignment Game, Computing Edit Distance, Reconstructing an Optimal Allignment	
6		**05.3-Dynamic Programming-Knapsack** Problem Overview, Knapsack with Repetitions, Knapsack without Repetitions, Final Remarks **05.4-Dynamic Programming-Placing Parentheses**Problem Overview, Subproblems, Algorithm, Reconstructing a Solution	
7		**06-Text Processing** Abundance of Digitized Text, Pattern-Matching Algorithms, Dynamic Programming, Text Compression and the Greedy Method, Tries	
8		**Midterm Exam**	
9		**08-Graph Algorithms-Graphs and Traversals (1-4)** **08.1-Graph Algorithms-Graphs and Traversals** Graphs, Data Structures for Graphs, Graph Traversal Algorithms, Depth-First Search (DFS), Breadth-First Search (BFS) **08.2-Graph Algorithms-Code Examples**	
10		**09-Graph Algorithms-Directed Graphs** Directed Graphs, Transitive Clousure, Directed Acyclic Graphs	
11		**10-Graph Algorithms-Shortest Paths and MSTs** Weighted Graphs, Dijkstra's Algorithm, Bellman-Ford Algorithm, Minimum Spanning Trees (MSTs), Prim-Jarnik's Algorithm, Kruskal's Approach, Disjoint Partitions and Union-Find Structures	

12		**11-Graph Algorithms in AI Search Problems-Maze Traversal (4-4)** **11.1-Graph Algorithms-Uninformed Search using DFS and BFS** What is Artificial Intelligence (AI)?, Formulating AI Search Problems, State Space Graphs and Search Trees, Tree Search Algorithms, Depth-First Search (DFS), Breadth-First Search (BFS) **11.2-Graph Algorithms-Maze Traversal using DFS and BFS** Search Algorithms: DFS and BFS, An Approach for Implementing Search Tree, Revised Approach Using DFS and a Stack, Basic Tree Search using BFS and a Queue, Maze Path Finding Example using DFS and BFS, Python Maze Example	
13		**12-Constraint Satisfaction Problem (CSP)-Backtracking Search (1-2)** Defining Contraint Satisfaction Problems, Solving CSPs, Backtracking Search for CSPs, Backtracking Examples	
14		**13-Constraint Satisfaction Problem (CSP)-Improving Backtracking (2-2)** Filtering: Forward Checking and Arc Consistency , Ordering: MRV and LCV, The Structure of Problems, Iterative Improvement, Local Search for CSPs	
15		**Project Demonstrations / Midterm Exam II**	
		Final Exam	

GRADING
Practical Part:

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

Theoretical Part:

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