NJIT

Department of Computer Science

CS 435-001: Advanced Data Structures and Algorithm Design-Fall 2025

Class Location: CULM LECT 2 **Recitation:** KUPF 210	**Class Time:** TR, 1:00 PM - 2:20 PM **Recitation:** F, 1:00 PM - 2:00 PM

Instructor: Marzieh Eskandari

Email: marzieh.eskandari@njit.edu

Zoom: https://njit-edu.zoom.us/my/meskandari

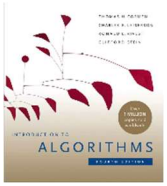
Office: GITC 4313	**Office Hours:** TR 2:30 PM- 3:20 PM

Prerequisites:

1. CS 241 (Discrete Math)
2. CS 288 (Intensive Programming)

Textbook:

T.H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein,
Introduction to Algorithms,
Third Edition¸ MIT Press, 2009
ISBN-13: 978-026203384-8



Evaluation:

Assignments: 25%
Attendance: 5%
Class Activity: +5% (Extra)
Midterm Exam 1 (**Oct 14th**): 20%
Midterm Exam 2 (**Nov 13th**): 20%
Final Exam (**Dec**): 30%

Note 1: NJIT Picture ID required for all exams. All exams are closed books and closed notes.

Course Description:

This is a senior-level course on data-structures and algorithms, with an emphasis on algorithm design techniques and analysis of algorithms. Topics include analysis of algorithms, worst-case and average-case analysis, induction, recursion, recurrence relations, divide-and-conquer design technique, priority queues, hash tables, binary-search trees, balanced search trees (AVL trees and red-black trees), sorting and selection algorithms; other design techniques such as greedy-method and dynamic-programming, and graph algorithms.

Course Objectives (what you are expected to get out of this course):

1. Master basic techniques for analyzing algorithm performance.
2. Understand recurrence relations and their role in algorithm analysis.
3. Compare and analyze sorting algorithms and their complexities.
4. Learn about search trees, balanced trees, and hashing techniques.
5. Understand fundamental graph algorithms and their applications.
6. Explore key algorithm design paradigms, including divide-and-conquer, greedy and dynamic programming.

Grading:

The grading scale is: A: 93–100%, B+: 85–92%, B: 77–84%, C+: 70–76%, C: 60–69%, D: 50–59% F: 0–49%.

Academic Integrity:

Academic Integrity is the cornerstone of higher education and is central to the ideals of this course and the university. Cheating is strictly prohibited and devalues the degree that you are working on. As a member of the NJIT community, it is your responsibility to protect your educational investment by knowing and following the academic code of integrity policy that is found at: http://www5.njit.edu/policies/sites/policies/files/academic-integrity-code.pdf

Schedule of Assignments & Due Dates:

Assignment	Subject	Date
Homework 1	Lectures 1-4	Sep 11-Sep 18
Homework 2	Lectures 5-8	Sep 30- Oct 7
Midterm Exam 1	**Lectures 1-9**	**Oct 16**
Homework 3	Lectures 9-13	Oct 23-Oct 30
Homework 4	Lectures 14-18	Nov 1-Nov 8
Midterm Exam 2	**Time complexity+ Lectures 10-17**	**Nov 13**
Homework 5	Lectures 19-22	Nov 20-Nov 27
Homework 6	Group Assignment- Mini Project	Nov 29-Dec 12
Final Exam	**Time complexity+ Lectures 18-28**	**Dec**

Course Outline:

Lecture(s)	Date	Topic
1	Sep 2	Introduction: Percolation Problem
2,3,4	Sep 4, 9, 11	Algorithm Analysis Techniques, Asymptotic Notation.
5,6	Sep 16, 18	Recursive Algorithms, Recurrence Relations
7,8,9	Sep 23, 25, 30	Trees, Tries
	Oct 2	**Wellness day- No class**
10, 11	Oct 7, 9	Priority Queues, Heaps, Heapsort
12	Oct 14	Review of Sorting Algorithms: Insertion-Sort, Bubble-Sort, Selection-Sort
13	**Oct 16**	**Midterm Exam 1: Lectures 1-9**
14	Oct 21	Merge-Sort, Quicksort. Integer Sorting: Bucket-Sort, Radix Sort (MSD, ...), Counting Sort
15	Oct 23	Dictionary ADT and Hash Tables
16, 17	Oct 28, 30	Balanced Search Trees: AVL and Red-Black trees
18	Nov 4	Graphs: Representations and Traversals, Topological Sorting
19	Nov 6	Divide-and-Conquer: Mergesort, Strassen's Matrix Multiplication
20	Nov 11	Divide-and-Conquer: Large Integer Multiplication (Karatsuba), Median of Medians
21	**Nov 13**	**Midterm Exam 2: Time Complexity+ Lectures 10-17**
22, 23, 24	Nov 18, 20, 25	Greedy method: Fractional Knapsack, Task Scheduling, Huffman Coding, MST Algorithms (Kruskal, Prim), Single-Source-Shortest-Paths (Dijkstra)
	Nov 27	**Thanksgiving- No class**
25, 26,27	Dec 2, 4, 9	Dynamic Programming: Introduction, Binomial coefficients, All-Pairs-Shortest-Paths (Floyd-Warshall), Matrix Chain Multiplication, Optimal Binary Search Tree
28	Dec 11	Review, Problem Solving Prep: Final Exam Readiness