NATIONAL UNIVERSITY
of Computer & Emerging Sciences, Lahore

Department of Computer Science

CS302 – Design and Analysis of Algorithms

SPRING 2023

Instructor Name: Aamina Batool
Email address: aamina.batool@lhr.nu.edu.pk
Office Location/Number: tbd
Office Hours: tbd

Course Information

Program: BSCS	**Credit Hours:** 3	**Type:** Core	
Pre-requisites: Data Structures			
Class Meeting Time:	**BCS-4G**	Tuesdays, Thursdays 01:00 PM to 02:20 PM	
Class Venue:			
	BCS-4G	E&M - 5	

Course Description:

The objective of this course is not to fill your brains with every algorithm that you would ever need. One of the aims of this course is to teach you to reason about algorithms and describe them. In addition, many known algorithms to solve known problems will be taught. At the end of the course, you should be able to choose an appropriate algorithm from a set of algorithms for a given problem.

Course Learning Outcomes (CLOs):
1. **Design** algorithms using different algorithms design techniques i.e., Brute Force, Divide and Conquer, Dynamic Programming, Greedy Algorithms and apply them to solve problems in the domain of the program
2. **Analyze** the time and space complexity of different algorithms by using standard asymptotic notations for recursive and non-recursive algorithms
3. **Evaluate** the correctness of algorithms by using theorem proving or executing test cases

Course Textbook

- *Introduction to Algorithms* by Cormen, Leiserson, Rivest, and Stein, 3rd Ed., MIT Press, 2001.

Additional references and books related to the course:

- Jon Kleinberg, Éva Tardos, Algorithm Design, Pearson/Addison-Wesley
- Sanjoy Dasgupta, Christos Papadimitriou, Umesh Vazirani, Algorithms, McGraw-Hill Education
- *Algorithms in C++* by Robert Sedgewick, Addison-Wesley, 1992.
- *Data Structures and Algorithms* by Aho, Hopcroft, and Ullman.

Weekly Schedule

Lectures	Description	Chapters of Text
Week -1	The role of algorithms in computers, Asymptotic functions and notations (Big-oh, big-omega, theta) best and worst case time complexity	1, 2, 3
Week – 2, 3, 4	Divide and Conquer (maximum subarray sum, counting inversions, quicksort, merge sort) + Solving recurrences	2, 3, 6
Week – 5	Lower bound for comparison based sorting, Sorting in linear time: Count Sort, radix sort	8
	Midterm – I	
Week – 6,7	Dynamic Programming (maximum subarray, rod cutting, longest common subsequence, binary knapsack)	15
Week – 8, 9	Greedy Algorithms (Activity selection, fractional knapsack and Huffman codes)	16
Week – 10	Introduction to graphs (revision of BFS, DFS) and their application (topological sort, strongly connected components)	22
	Midterm – II	
Week – 11	Minimum Spanning Trees (MST)(Prim's Algorithm and Kruskal's Algorithm)	23
Week – 12, 13	Shortest Path Algorithms (Dijkstra's Algorithm, Bellman-Ford and Floyd Warshall Algorithm	24
Week 14	Approximation Algorithms/ NP Hard Problems	
	Final Exam	Comprehensive

Grading Criteria

1. Quizzes (10%)
2. Assignments (15%)
3. Midterm Exams (30%)
4. Final Exam (45%)

Grading Policy

Absolute Grading

Course Policies

1. Quizzes will be announced.
2. No makeup for missed quizzes and assignments.

Academic Integrity: All work MUST be done individually. Any copying of work from other person(s) or source(s) (e.g., the Internet) will automatically result in at least an F grade in the course. It does not matter whether the copying is done in an assignment, quiz, midterm exam, or final exam, it will be considered equally significant.