Aror University of Art, Architecture, Design & Heritage Sukkur.

BS(Artificial Intelligence)

Fall-2025

Analysis of Algorithm (Theory)

Course Title: Analysis of Algorithm
Course Code: CSC-331
Credit Hours: (3,0)
Course Instructor: Abdul Khalique
Electronic mail: akhalique.faculty@aror.edu.pk

Assessment:

S. No	**Assessment Activities**	Percentage	Total Activities
1.	Sessional: Quizzes/ Assignments (Quizzes, Assignments, & Test)	30%	4
2.	Mid Term Exam	30%	1
3.	Final Exam	40%	1

Course Objectives

The main objective of this course is to make the student familiar with subjects concerning algorithm complexity. By the end of the semester, the student should be able to: (1) recognize the use of several design techniques (greedy, divide-and-conquer, dynamic programming) and use these methods to solve simple Problems. (2) write and solve recurrence relations for recursive algorithms. (3) Determine asymptotic growth rates for algorithms.

Course Description

This course applies design and analysis techniques to numeric and nonnumeric algorithms which act on data structures. Design is emphasized so that the student will be able to develop new algorithms. Analysis of algorithms is concerned with the resources an algorithm must use to reach a solution. Topics include introduction to algorithm, asymptotic complexity, sorting and searching, divide and conquer, greedy graph algorithms, dynamic programming, data compression, backtracking, branch and bound

TEXT/REFERENCE BOOKS:

Introduction to Algorithms, Thomas H. Cormen Charles E. Leiserson Ronald L. Rivest Cliff, 3rd Edition, MIT Press,2009

SYLLABUS:

Week	Topics
1	what are Algorithms, Why do you need to study Algorithm, What kinds of problems are solved by Algorithms, Algorithms as Technology, correctness and generality of Algorithms, Major Factors in Designing an Algorithm
2	Major Assumptions in analyzing an algorithm, RAM Model Mathematical analysis of non-recursive algorithms, Complexity Classes and Insertion sort, Merge Sort
3	Asymptotic notations; Big O, Big Ω, little o, little-ω

4	Correctness of Recursive Algorithm, Mathematical Induction, Growth Rate of functions, Asymptotic Notations
5	Pre-Condition, Post Condition, Partial Correctness of an Algorithm, Total Correctness of An Algorithm Loop Invariant, Correctness of an Iterative Algorithm
6	Time and Space Tradeoffs in algorithm, Brute force Algorithms and their Analysis
7	Mathematical Analysis of recursive algorithms, Substitution Method, Tree Methods
8	Divide and conquer Algorithm,
9	Mid Term
10	Merge Sort and its Worst-case analysis,
11	Best and Average Case Analysis of merge sort, Quick Sort and its Worst, Best and Average Case Analysis, Heap Sort
12	Dynamic Programming, elements of Dynamic Programming, Comparison of Dynamic Programming with divide and Conquer , Knapsack Problem solving using Dynamic Programming
13	Longest Common Subsequent problem solving through Dynamic Programming, Shortest Path finding through Dynamic Programming
14	Graphs, and Graph Terminology, Graph Greedy Algorithms,
15	MST, Prims Algorithm
16	Kruskal Algorithm, Dijkstra Algorithm
17	Revision

COURSE LEARNING OUTCOMES (CLOs):

CLO	DESCRIPTION	Domain	BT Level
1	Prove the correctness and analyze the running time of the basic algorithms for those classic problems in various domains.	C	1
2	Apply algorithms and design techniques to solve problems.	C	2
3	Analyze the complexities of various problems in different domains.	C	3

Aror University of Art, Architecture, Design & Heritage Sukkur.

*BT=Bloom's Taxonomy, C=Cognitive domain, P=Psychomotor domain, A=Affective domain

MAPPING COURSE LEARNING OUTCOMES (CLOs) ANDPROGRAM LEARNINGOUTCOMES (PLOs)

	PROGRAM LEARNING OUTCOMES (PLOs)									
	1	**2**	3	4	5	**6**	**7**	**8**	9	**10**
CLO. 1			X							
CLO. 2	X									
CLO. 3		X								

PROGRAM LEARNING OUTCOMES (PLOs):

S#	**Program Learning Outcomes (PLOs)**	**Computing Professional Graduate**
1	Academic Education	To prepare graduates as computing professionals
2	Knowledge for Solving Computing Problems	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
3	Problem Analysis	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
4	Design/ Development of Solutions	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
5	Modern Tool Usage	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
6	Individual and Team Work	Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings.