MANGALORE UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

CSH 402: ALGORITHMICS		
Hours/Week: 4 **Credits : 4**		**I.A. Marks: 30** **Exams. Marks: 70**

Course Outcomes:

CO1: Argue the correctness of algorithms using inductive proofs and invariants.

CO2: Analyze worst-case running times of algorithms using asymptotic analysis.

CO3: Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.

CO4: Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.

CO5: Describe the greedy paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize greedy algorithms, and analyze them.

CO6: Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate. Synthesize new graph algorithms and algorithms that employ graph computations as key components, and analyze them.

CO7: Explain the different ways to analyze randomized algorithms (expected running time, probability of error). Recite algorithms that employ randomization. Explain the difference between a randomized algorithm and an algorithm with probabilistic inputs.

	UNIT-I	**12 Hrs**

Introduction: Algorithms, performance analysis-time complexity and space complexity, O-notation, Omega notation and Theta notation, Review of basic data structures , priority queues- , heaps, definition, insertion and deletion, application-heap sort, Introduction to Skip List, skip list representation, operations- insertion, deletion and searching , Hashing, hash table representation, hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing and comparison of hashing and skip lists.

	UNIT-II	12 Hrs.
Search Trees: Binary Search Trees, definition, ADT, implementation, operations-searching, insertion and deletion, Balanced search trees- AVL trees, definition, height of an AVL tree, representation, operations-insertion, deletion and searching. Introduction to Red – Black trees and Splay Trees, B-Trees, insertion, deletion and searching, Comparison of Search Trees.		
	UNIT-III	**12 Hrs.**
Divide and Conquer: General Method – Binary Search – Finding Maximum and Minimum – Merge Sort ,**Greedy method**: General method, Minimum cost spanning trees, Job sequencing with deadlines, **Backtracking:** General Method – 8 Queens problem – sum of subsets – graph coloring – Hamiltonian problem – knapsack problem.		
	UNIT-IV	**12 Hrs.**
Dynamic Programming: General method, Optimal binary search trees, 0/1 knapsack problem, Travelling sales person problem. **Graphs:** Graph Traversals – Connected Components – Spanning Trees – Biconnected components – Branch and Bound: General Methods (FIFO & LC) – 0/1 Knapsack problem – Introduction to NP-Hard and NP-Completeness.		

REFERENCE BOOKS:

1. Mark A. Weiss, "Data structures and Algorithm analysis in C++(Java)", Fourth Edition, PHI ,2013
2. Michael T. Goodrich, R. Tamassia and D. Mount "Data structures and Algorithms in C++", Wiley student edition, John Wiley and Sons.
3. Data Structures and Algorithms in C++, Second Edition, Adam Drozdek, Vikas Publishing House, Thomson International Student Edition.
4. **[4]** Ellis Horowitz, SartajSahni and SanguthevarRajasekaran, "Computer Algorithms/ C++", Second Edition, Universities Press, 2007.
5. Horowitz and Sahni, and Rajashekaran, Fundamentals of Computer Algorithms, University Press, 2nd Edition, Galgotia Publications, 2007
6. AnanyLevitin ,Introduction to the Design & Analysis of Algorithms, Pearson Addison-Wesley, 2007 .