

1151CS106	DESIGN AND ANALYSIS OF ALGORITHM	3	0	0	3
-----------	----------------------------------	---	---	---	---

**Course Category:** Program Core

**A. Preamble :**

For an engineer, problem solving is not about just solving a problem somehow but about solving the problem in the most effective and efficient way. Two key skills that a software professional needs are (1) to choose suitable data structures to store the information part of the problem, and (2) use of efficient algorithms for developing a programming solution of a given problem. Selection of a particular data structure greatly influences the characteristics of the obtained solution that include efficiency (performance, or speed), space (memory) requirements, scalability, reuse, and robustness (or reliability). The other equally important skill is to choose a suitable problem solving technique to apply to a particular problem. Acquiring these skills, greatly enhances the problem solving skills of the learner.

**B. Prerequisite Courses:**

Sl. No	Course Code	Course Name
1	1151CS102	Data Structures

**C. Related Courses:**

Sl. No	Course Code	Course Name
1	1156CS601	Minor Project
2	1156CS701	Major Project

**D. Course Outcomes :**

Upon the successful completion of the course, students will be able to:

CO Nos.	Course Outcomes	Level of learning domain (Based on revised Bloom's)
CO1	Explain various asymptotic notations and Compute the efficiency of given algorithms	K2
CO2	Apply the brute force technique to solve the given problem	K3
CO3	Use DAC technique to solve a given problem.	K3
CO4	Compute optimum solutions for the given problem.	K3
CO5	Apply B&B and B&T technique to solve combinatorial problem	K3
CO6	Discuss the improvement of computational efficiency using iterative approaches	K2

**E. Correlation of COs with POs :**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	L	L								
CO2			M	L	L							
CO3		M	M	M		L						
CO4		M	M	M	L	L						
CO5		M	M	L	L		L					
CO6		M		M	L	L						

## **F. Course Content:**

### **UNIT I INTRODUCTION**

**9**

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types – Fundamentals of the Analysis of Algorithm Efficiency – Analysis Framework – Asymptotic Notations– Mathematical analysis for Recursive and Non-recursive algorithms.

### **UNIT II BRUTE FORCE AND DIVIDE-AND-CONQUER**

**9**

Brute Force: Closest-Pair Problems- Exhaustive Search - Traveling Salesman Problem - Knapsack Problem - Assignment problem. Divide and conquer methodology: Merge sort – Quick sort – Binary search.

### **UNIT III DYNAMIC PROGRAMMING AND GREEDY TECHNIQUE**

**9**

Dynamic Programming: Computing Binomial Coefficient Warshall's and Floyd' algorithm – Knapsack Problem. Greedy Technique: Prim's algorithm- Kruskal's Algorithm- Dijkstra's Algorithm.

### **UNIT IV BACKTRACKING AND BRANCH & BOUND**

**9**

Backtracking: n-Queens problem-Hamiltonian Circuit Problem - Branch and Bound: Assignment problem-Knapsack Problem- Traveling Salesman Problem

### **UNIT V ITERATIVE IMPROVEMENT AND LIMITATIONS OF ALGORITHM POWER**

**9**

The Maximum matching in bipartite graph. Limitations of Algorithm Power--Decision Trees- P, NP and NP-Complete Problems.

**TOTAL: 45Periods**

## **G. Learning Resources**

### **i.Text Books:**

1. AnanyLevitin, “Introduction to the Design and Analysis of Algorithms”, Third Edition, Pearson Education, 2012.

### **ii. REFERENCES:**

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, Third Edition, PHI Learning Private Limited, 2012.
2. Donald E. Knuth, “The Art of Computer Programming”, Volumes 1& 3 Pearson Education, 2009.
3. “Fundamentals of Computer Algorithms” by Ellis Horowitz, SartajSahmi, SanguthevarRajasekaran, University Press, Second Edition 2008.

### **iii. Online Resources:**

1. <http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html>
2. <http://nptel.ac.in/courses/106101060/>
3. <https://www.coursera.org/course/algo>