

<b>Devi Ahilya University, Indore, India Institute of Engineering &amp; Technology</b>				<b>III Year B.E. (Information Technology (Full Time))</b>			
<b>Subject Code &amp; Name</b>	<b>Instructions Hours per Week</b>			<b>Credits</b>			
<b>ITR6C2 Design and Analysis of Algorithms</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Total</b>
<b>Duration of Theory Paper: 3 Hours</b>	<b>3</b>	<b>1</b>	<b>1</b>	<b>3</b>	<b>1</b>	<b>1</b>	<b>5</b>

### **Learning Objectives:**

1. Reinforce basic design concepts (e.g., pseudocode, specifications, top-down design)
2. Knowledge of algorithm design strategies
3. Familiarity with an assortment of important algorithms
4. Ability to analyze time and space complexity

### **Prerequisite:**

The students are required to have familiarity with the following data structures: Arrays and linked lists, Stacks and queues, Graphs and trees, binary search trees, height balancing, Heaps and priority queues

## **COURSE CONTENTS**

### **UNIT-I**

**Introduction to Algorithms:** Notion of algorithms, properties, important areas of research in connection with the study of algorithms, Types of algorithms; Analysis- best case, worst case, and average case. Performance issues - Time and space complexity; Asymptotic analysis. Mathematical preliminaries; functions & their growth rates; Recurrence relations, Methods for solving recurrences.

### **UNIT-II**

**Selected Algorithms for Sorting, Searching and matrix multiplication:** Elementary sorting techniques: Selection, Bubble, and Insertion sorts; Advanced sorting techniques: Heap, Merge and Quick sorts; Radix & Bucket sorts. Searching techniques: Linear and binary search; Searching minimum and maximum elements. Divide-and-Conquer strategy, Strassen's matrix multiplication.

### **UNIT-III**

**Greedy Method and Dynamic Programming:** Algorithms design techniques based on Greedy Method and Dynamic programming. Illustration of these strategies using appropriate examples including Knapsack problem, optimal storage on tapes, finding shortest path, all pairs shortest path, finding minimum cost spanning trees, and Matrix chain multiplication problem.

### **UNIT-IV**

**Backtracking, Branch-and- Bound, and String Matching:** Backtracking and Branch-and-Bound algorithm design techniques, Illustration of these techniques using appropriate examples like Queens Problem, subset sum problem, traveling salesperson problem, etc.

Introduction to string matching problem, Applications, String matching algorithms: Naive algorithm, Rabin-Karp, Knuth-Morris-Pratt, Boyer-Moore, etc.

### **UNIT-V**

#### **The Theory of NP-Completeness:**

Non-deterministic Algorithms: Introduction. Nondeterministic Complexity, Decision and optimization problems, Tractable and Intractable Problems, Computational Classes: – P, NP, NP-Complete, and NP-Hard; reducibility, Selected NP-Complete and NP-Hard problems: Hamiltonian cycle, Traveling Salesperson (TSP). Satisfiability, Clique problems, etc.

#### **Learning Outcomes:**

Students who have completed this course should be able to:

1. Apply design principles and concepts to algorithm design
2. Have the mathematical foundation in analysis of algorithms
3. Understand different algorithmic design strategies
4. Analyze the efficiency of algorithms using time and space complexity theory

**Assessment methods of all of the above:** quizzes, exams, assignments, practicals

#### **Books Recommended:**

1. Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms, Second Edition, MIT Press/McGraw-Hill, 2001.
  2. Michael T Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Wiley, 2006.
  3. Jon Kleinberg and ÉvaTardos, Algorithm Design, Pearson, 2005.
  4. E. Horowitz, S. Sahni, S Rajasekaran, Computer Algorithms, Galgotia Publications.
  5. Saara Base, Computer Algorithms: Introduction to Design and Analysis, Addison Wesley, 2/e, 1988.
  6. Knuth, D, The art of computer programming, Vols. 1-2-3, Addison Wesley 1968-73.
  7. A V Aho, J E Hopcroft& J D Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley, 1974.
  8. Vijay V Vazirani, Approximation Algorithms, Springer-Verlag, 2001
-