19IT302 DESIGN AND ANALYSIS OF ALGORITHMS

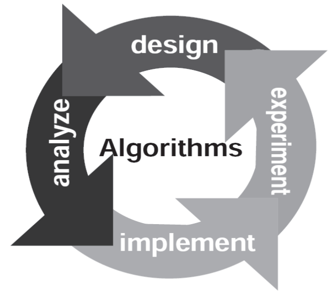
Hours Per Week :

L	T	P	C
2	-	4	4

Total Hours :

L	T	P
30	-	60

WA/RA	SSH/HSH	CS	SA	S	BS
5	40	5	20	5	2



SOURCE:https://www.tes.com/lessons/rcrxkMCBbphf4w/cape-cs1m2-4-ways-of-representing-algorithms

PREREQUISITE COURSES:

Data Structures; C/ C++/ Python/ Java/ Any one Programming language

COURSE DESCRIPTION AND OBJECTIVES:

This course offers insight into the time and space complexities of various algorithms, designof algorithms using divide and conquer, greedy, dynamic, branch and bound, and backtracking approaches. The objective of this course is to design and analyse the algorithms for their time and space complexities and to understand problems such as 0/1 knapsack,shortest path, minimum spanning tree, matrix multiplication, graph coloring, n-queens and travelling salesman problem.

COURSE OUTCOMES*:*

Upon completion of the course, the student will be able to achieve the following outcomes:

COs	Course Outcomes	POs
1	Understand different algorithmic design strategies like divide & conquer,Greedy, dynamic programming and backtracking etc.	-
2	Apply various design algorithms to solve a given problem.	1,9
3	Analyze the efficiency of a given algorithm using time and space complexity theory.	2,9
4	Investigate which algorithm is efficient to solve a given novel problem.	4,9
5	Synthesize new algorithms for solving given problems based on different design strategies like divide and conquer, greedy, dynamic programming and backtracking techniques and analyze them.	3,9

SKILLS:

- ✓ *Develop algorithms for solving problems using divide and conquer, greedy, dynamic programming and backtracking techniques.*
- ✓ *Analyze the given algorithm with respect to space and time complexities and compare with other algorithms.*
- ✓ *Application of existing algorithms to solve real world problems.*

UNIT - I **L-6**

INTRODUCTION: Algorithm, Pseudo-code for expressing algorithms, Performance analysis - Space and Time complexity; Asymptotic notation - Big oh notation, Omega notation, Theta notation and Little oh notation; Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method, Masters theorem.

DIVIDE AND CONQUER: General method, Applications - Binary search, Quick sort, Merge sort, and Stassen's matrix multiplication.

UNIT - II **L-6**

GREEDY METHOD: Applications - Job sequencing with deadlines, Knapsack problem, Minimum cost spanning trees, Single source shortest path.

Disjoint sets - Disjoint set operations, Union and find algorithms, Graph representations. Connected components and Bi-connected components.

UNIT - III **L-6**

DYNAMIC PROGRAMMING: General method, Applications - Optimal binary search trees, Matrix chain multiplication, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem.

UNIT - IV **L-5**

BACKTRACKING: General method, Applications – n-queen problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles.

UNIT - V **L-7**

BRANCH AND BOUND PROBLEMS: General method, Applications - Traveling sales person problem, 0/1 knapsack problem, LC Branch and Bound solution, FIFO Branch and Bound solution.

P, NP, NP - Hard and NP-Complete: Basic concepts, Non deterministic algorithms, NP – Hard and NP complete classes, Cook's theorem, Randomized algorithms.

LABORATORY EXPERIMENTS

LIST OF EXPERIMENTS **TOTAL HOURS: 60**

Language: C / C++ / Java / Python / Any other programming language of student's choice.

1. Sort a given set of elements using the following methods and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.

   a.Quick sort   b. Merge sort   c. Bubble sort

2. Search for a given element/ elements using the following methods and determine the time required to search the given element. Repeat the experiment for different values of n. The elements can be read from a file or can be generated using the random number generator.

   a.Linear Search   b. Binary Search

3. Implement the following:

   a. Represent the given graph using sequential representation.

   b. Find the degree of all the vertices.

   c. List all the isolated vertices.

4. Implement the following using divide and conquer approach

   a.To multiply two given square matrices.

   b.To multiply two given square matrices using starssen's matrix multiplication.

5. State the problem of Job sequencing with deadlines and write a program to solve it using Greedy method.

6. State Knapsack problem and write a program to solve the knapsack problem using Greedy approach.

7. Define minimum spanning tree and write a program to find minimum spanning tree for a given undirected graph using any algorithm of your choice.[Prims/ Krushkals].

8. State all pairs shortest path problem and write a program to solve it using dynamic programming.

9. Explain optimal binary search tree using an example and Write a program to find optimal binary search tree using dynamic programming.

10. State the problem of Matrix chain multiplication and write a program to find optimal order of matrix chain multiplication problem using dynamic programming.

11. State n-queens problem and write a program to solve the same using backtracking approach.

12. State the problem of sum of subsets and write a program to solve it using backtracking approach.

13. Write a program to solve knapsack problem using Branch and Bound.

TEXT BOOKS:

1. Ellis Horowitz, Satraj Sahni and Rajasekharam, “Fundamentals of Computer Algorithms”, 2nd edition, Galgotia publications, 2006.

2. Thomas H.Coremen,Charles E.Leiserson and Ronald L.Rivest, “ Introduction to Algorithm”, 2nd edition, Clifford Stein, 2014.

REFERENCE BOOKS:

1. Anony Levitin, “Introduction to Design and Analysis of Algorithms”, 3rd edition, Pearson Education, 2016.

2. Donald E.knuth, “The Art of Computer Programming”, Volume 3, 2nd edition, Addisonwesley Longman Inc,1998.

3. Ronald L.Graham,Donald E.Knuth and Oren Patashnik, “Concrete Mathematics”, 2nd edition, Addison-wesley Publishing Company,1998.