SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN
(AUTONOMOUS)
Approved by AICTE & Affiliated to JNTUK, Kakinada Accredited
with ‘A’ Grade by NAAC & NBA
Vishnupur, Bhimavaram, West Godavari Dist. – 534 202, Andhra Pradesh, India.

Student Notebook	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I Semester
Subject	DESIGN AND ANALYSIS OF ALGORITHMS
Regulation	R18
Subject Code	UGIT5T0118

VISHNU
UNIVERSAL LEARNING

Vision
Transform the society through excellence in education, community empowerment and sustained environmental protection.

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN:: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Vision:

To establish unique identity by development of high quality IT engineers and technological resources for contributing to the economic and social development of the Nation at large and region in particular.

Mission:

- To provide for the holistic development of undergraduate students in the Information Technology.
- To prepare students for careers in industry or to pursue advanced graduate studies to get involved in research activities.
- To provide a teaching environment that emphasizes continuous learning and inculcates professional ethics.
- To establish centers of excellence in various domains.

Program Educational Objectives (PEOs):

PEO 1. Graduates will be leaders in academia, industry and research pursuit through strong Knowledge in core and application domain, that develops the ability to solve real world problems individually and in team.

PEO 2. Graduates will continue to learn and adapt in a world of constantly evolving technology.

PEO 3. Graduates will have deep awareness of ethical responsibilities in their profession and towards the society.

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN:: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Program Outcomes (POs):

PO 1. An ability to use principles and methods of sciences, mathematics and engineering disciplines to solve technical problems.

PO 2. An ability to analyze a problem, identify and define the computing requirements appropriate to its solution.

PO 3. An ability to design, implements, and evaluate a computer-based system, process, component or program to meet desired needs.

PO 4. An ability to design and conduct experiments, as well as to analyze and interpret data.

PO 5. An ability to use current techniques, skills, and modern tools necessary for computing practice.

PO 6. The education necessary to understand the impact of engineering solutions in the economic, environmental, and societal context.

PO 7. An understanding of impact of engineering solutions on the society and awareness of contemporary issues.

PO 8. An ability to practice professional and ethical responsibilities.

PO 9. The ability to learn, unlearn and relearn technologies, both as a individual and within a collaborative team.

PO 10. An ability to communicate and function effectively in teams to accomplish a common goal.

PO 11. Recognize the need for and an ability to engage in continuing professional development.

PO 12. An understanding of engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects.

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Syllabus	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I-Semester
Subject	DESIGN AND ANALYSIS OF ALGORITHMS
Regulation	R18
Subject Code	UGIT5T0118

UNIT-I

Introduction:

Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, probabilistic analysis, Amortized analysis.

UNIT-II

Disjoint sets and Divide and Conquer: Disjoint set operations, Union and find algorithms, Spanning trees. Divide and Conquer methodology, applications-Binary search, Quick sort, Merge sort, Multiplication of large integers, Strassen's matrix multiplication.

UNIT -III

Greedy method: General methodology, applications- knapsack problem, Minimum cost spanning trees, Single source shortest path problem,

UNIT -IV

Dynamic Programming: General methodology, applications-0/1 knapsack problem, Optimal binary search trees, All pairs shortest path problem, Traveling sales person problem.

UNIT -V

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring,Hamiltonian cycles. Branch and Bound methodology, applications, LC branch and bound, 0/1 knapsack problem: LC Branch and bound solution, FIFO branch and bound solution, Travelling sales person problem.

UNIT –VI

NP-hard and NP-Complete problems: basic concepts, non deterministic algorithms, NP-hard and NP-complete classes, list of NP-hard and NP-complete problems,Cooks theorem

TEXT BOOKS:

T1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
T2. ParagHimans hu Dave, Himanshu BhalchandraDave, Design and Analysis of Algorithms, Pearson Publication.
T3. M.T. Goodrich, Roberto Tamassia Algorithm Design, Foundation, Analysis and Internet Examples, Wiley.

REFERENCES:

R1. Introduction to Algorithms, second edition,T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein,PHI Pvt. Ltd.
R2. R C T Lee, Hang and TT Sai, Introduction to Design and Analysis of Algorithms , A strategic approach,TMH
R3. Allen Weiss, Data Structures and Algorithms Analysis in C++, 2^{nd} Edn, Pearson Education.
R4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft,Pearson education.
R5. Richard Johnson Baugh, and Marcus Schaefer, Algorithms, Pearson Education.

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN: BHIMAVARAM
(AUTONOMOUS)
LESSON PLAN

COURSE: III B.Tech	**BRANCH** : IT
CLASS : III/ISem. Section	**YEAR** : 2020-21
FACULTY:	**DESIGNATION:**
SUBJECT: Design and Analysis of Algorithms	**SUBJECT CODE:** UGIT5T0118

.

Prerequisites: Familiarity with Problem Solving Skills, Discrete Mathematics and Data Structures.

Course Objectives: The students will learn the following:

1. Relate the algorithm properties with mathematical approaches to design and analyze real time problems.
2. Make use of optimization techniques to solve complex problems in easy ways.
3. Ability to perform dynamic actions for the particular problem based on specific constraints.
4. Construction of state space tree in order to reduce the number of solutions and to find the optimal solution
5. Design elementary deterministic and randomized algorithms to solve computational problems

Course Outcomes: Upon the successful completion of the course, the student will be able:

CO 1 Understand the fundamentals of algorithmic design steps, performance analysis concepts and various algorithm design methods.

CO 2 Apply the algorithm design techniques to design efficient algorithms for different kinds of computing problems.

CO 3 Analyze the asymptotic performance of algorithms and write formal correctness proof for algorithms.

CO 4 Classify a problem as computationally tractable or intractable and discuss the strategies to address int

Lesson Plan:

S.No.	No. of hours	Date	Topic(s) planned	Reference (Books with page numbers)	Remarks
UNIT I -Introduction:					
1	2		Introduction Algorithm	T1-1-4 T2-1-18	
2	1		Pseudo code for expressing Algorithms	T1-5-9 T2-22-52	
3	1		performance Analysis	T1-14 T2-58	
4	1		Space complexity	T1-15-16 T2-88	

5	1		Time complexity	T1-17-25 T2-60-71	
6	2		Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation	T1-39-49 T2-76-87	
7	2		probabilistic analysis, Amortized analysis	T1-28-38	
UNIT II -Disjoint sets and Divide and conquer:					
8	1		Disjoint sets	T1-110	
9	1		Disjoint sets operations	T1-111	
10	1		Union algorithms	T1-115	
11	1		Find algorithm	T1-118	
12	1		spanning trees	T1-236 T2-401	
12	2		General method	T1-136-140 T2-262-263	
13	2		applications-Binary search	T1-145-153 T2-326-327	
14	2		Quick sort	T1-168-177 T2-269-275	
15	2		Merge sort	T1-159-167 T2-264	
16	1		Multiplication of large integers	R1-68	
17	1		Stassen's matrix multiplication	T1-192-194	
UNIT III - Greedy method:					
18	1		General method	T1-210-213 T2-372-374	
19	1		knapsack problem,	T1-218-222 T2-384-388	
20	3		Minimum cost spanning trees	T1-237-246 T2-401	
21	2		Single source shortest path problem.	T1-260-266 T2-407	
22	1		Huffman tress	T1-257-259	
UNIT-IV-Dynamic Programming:					
23	1		General method	T1-272-276 T2-455	
24	1		Applications-Matrix chain Multiplication	T2-488-496	
25	3		Optimal binary search trees	T1-293-302 T2-500-502	
26	1		0/1 knapsack problem	T1-305-312 T2-496	
27	2		All pairs shortest path problem	T1-284-288 T2-478-481	

28	2		Traveling sales person problem	T1-318-320 T2-486-487	
UNIT-V-Backtracking and Branch Bound:					
29	2		General method	T1-359-360 T2-517	
30	3		Applications-n-queen problem	T1-373-375 T2-522-523	
31	2		Sum of subsets problem	T1-377-379 T2-525-526	
32	2		Graph coloring	T1-380-384 T2-527	
33	2		Hamiltonian cycles.	T1-384-387 T2-531	
Branch and Bound:					
34	2		General method	T1-399-400 T2-543	
35	3		Travelling sales person problem	T1-422-430 T2-559	
36	2		0/1 knapsack problem-	T1-413-414 T2-562	
37	2		LC Branch and Bound solution	T1-414-417	
38	2		FIFO Branch and Bound solution	T1-417-420	
UNIT-IV NP-hard and NP-complete problems:					
39	1		Basic concepts	T1-514	
40	1		Non deterministic algorithms,	T1-515-520	
41	2		NP-hard and NP-complete classes,	T1-523-526	
42	2		list of NP-hard and NP-complete problems.	T1-527	

TEXT BOOKS:

T1.Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.

T2. ParagHimans hu Dave, Himanshu BhalchandraDave, Design and Analysis of Algorithms, Pearson Publication.

T3. M.T. Goodrich, Roberto Tamassia Algorithm Design, Foundation, Analysis and Internet Examples, Wiley.

REFERENCES:

R1. Introduction to Algorithms, second edition,T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein,PHI Pvt. Ltd.

R2. R C T Lee, Hang and TT Sai, Introduction to Design and Analysis of Algorithms , A strategic approach,TMH

R3.Allen Weiss, Data Structures and Algorithms Analysis in C++, 2nd Edn, Pearson Education.

R4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft,Pearson education.

R5. Richard Johnson Baugh, and Marcus Schaefer, Algorithms, Pearson Education.

Staff In-charge **Head of the Department**

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Descriptive Question Bank	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I-Semester
Subject	Design and Analysis of Algorithms
Regulation	R18
Subject Code	UGIT5T0118

UNIT-I

1. (a).Define an algorithm. What are the different criteria that satisfy the algorithms?

   (b). Explain how algorithms performance is analyzed? Describe asymptotic notations?

2. (a) What are the different techniques to represent an algorithm. Explain?

   (b) Give an algorithm to solve the towers of Hanoi problem.

   (a) Write an algorithm to find the sum of individual digits of a given number.

   (c) Explain the different looping statements used in pseudo code conventions.

3. (a) What is meant by recursion? Explain with example, the direct and indirect recursive algorithms.

   (b) List the advantages of pseudo code convention over flow charts.

UNIT-II

1. (a) Explain the binary search to find the elements 12,50,-2,45 from for the following set

   (3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 47)

   (b) Derive the time complexity for Quick.

2. (a) Draw the tree of calls of merge sort for the following set.

   (35, 25,15,10,45, 75, 85, 65, 55, 5, 20, 18)

   (b) Compare Quick sort algorithm performance with merge sort algorithm?

3. (a) Write the merge sort algorithm and Draw the tree of calls of merge for the following set of elements

   (20, 30, 10, 40, 5, 60, 90, 45, 35, 25, 15, 55)

   (b) Write an algorithm for quick sort by using recursive method.

4. (a)Explain the Disjoint sets with examples

   (b) Explain the union and find algorithms with examples.

5. (a) Explain the matrix multiplication for integer..

   (b) Explain the strassens matrix multiplication.

UNIT-III

1. (a) What is greedy method? Explain with example.

   (b) explain the Huffman tree with examples.

2. Explain the 0/1 knapsack problem. Consider the following instance of the knapsack problem n=3,m=20,

   (p1,p2,p3)=(25,24,15), and (w1,w2,w3)=(18,15,10).

3. Define minimum cost spanning trees. Explain them with suitable example.

4.(a)What are the observation that should made for finding the shortest paths by using Greedy.

   (b) Explain, how to find the minimum cost spanning tree by using Prim’s Algorithm.\

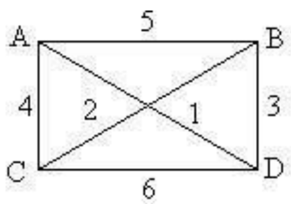
UNIT-IV

1. (a) Find the solution for the knapsack problem. When n=3,

   (W_1, W_2, W_3)=(18, 15, 10,) . (P1, P2, P3)= (25, 24, 15) and m=20.

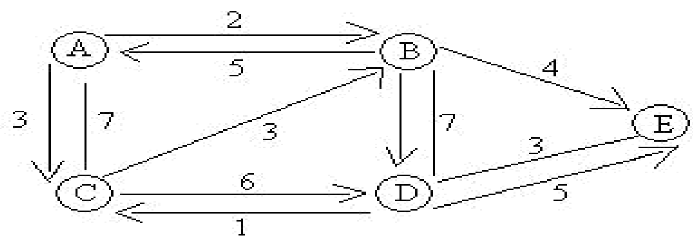
Explain the general concept of Dynamic programming.

2. (a) Find the shortest paths between all pairs of nodes in the following graph



   (b) What are the advantages of finding shortest paths and also explain the application areas.

3. Find the shortest path b/w all pairs of nodes in the following graph and explain with the suitable algorithm



4. (a) Discuss the dynamic programming solution for the problems of reliability design.

Define merging and purging rules in O/1 knapsack problem.

UNIT-V

1. a) Explain, how the Hamiltonian circuit problem is solved by using the backtracking concept.

Device a backtracking algorithm for m-coloring graph problem.

2.(a) Compare and contrast between Brute force approach Vs Back tracking.

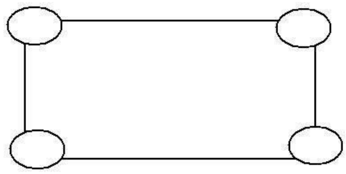
Suggest a solution for 8 queen's problem.

3. (a) Explain about graph coloring and chromatic number.

(b) For the graph given below, draw the portion of the state space tree generated by procedure MCOLORING
.

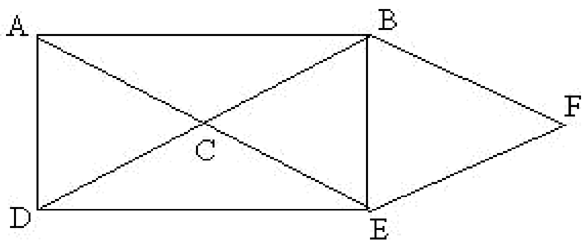
1 **2**

3 **4**



4. (a) Compare and contrast between Brute force approach and Backtracking.

(b) Find the Hamiltonian circuit in the following graph by using backtracking.



5.(a) Write FIFOBB algorithm for the 0/1 knapsack problem.

(b) Explain the general method of Branch and Bound.

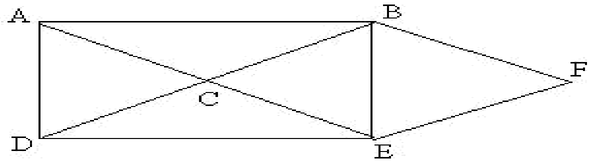
6. Apply the Branch and Bound algorithm to solve the TSP, for the following cost matrix.

∞	11	10	9	6
8	∞	7	3	4
8	4	∞	4	8
11	10	5	∞	5
6	9	5	5	∞

7. (a) Explain how the traveling salesperson problem is solved by using LC Branch and Bound.

(b) Write the general algorithm for Branch and Bound.

8. (a) Compare and contrast between Brute force approach and Backtracking.
   (c) Find the Hamiltonian circuit in the following graph by using backtracking.



9. What is traveling sales person problem? Solve the following sales person problem instance using branch and bound.

0	10	15	20
5	0	9	10
6	13	0	12
8	8	9	0

UNIT-VI

1. Explain the NP-hard and NP complete classes with examples.
2. Explain the NP-hard and NP-complete problems.

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN :: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment-1	
Department	Information Technology
Year / Semester	III B.Tech (IT) – ISemester
Subject	Design and Analysis of Algorithms
Regulation	R18
Subject Code	UGIT5T0118

UNIT-I

1. (a).Define an algorithm. What are the different criteria that satisfy the algorithms?

   (b). Explain how algorithms performance is analyzed? Describe asymptotic notations?
2. (a) What are the different techniques to represent an algorithm. Explain?

   (b) Give an algorithm to solve the towers of Hanoi problem.

.

UNIT-II

1. (a) Explain the binary search to find the elements 12,50,-2,45 from for the following set

   (3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 47)

   (b) Derive the time complexity for Quick.
2. (a) Draw the tree of calls of merge sort for the following set.

   (35, 25,15,10,45, 75, 85, 65, 55, 5, 20, 18)

   (b) Compare Quick sort algorithm performance with merge sort algorithm?

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN :: BHIMAVARAM
(AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment-2	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I-Semester
Subject	Design and Analysis of Algorithms
Regulation	R18
Subject Code	UGIT5T0118

UNIT 3	**UNIT 4**
Greedy method,prims Algorithm	Dynamic programming concepts,Travelling salesman problem

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN :: BHIMAVARAM (AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Assignment-3	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I Semester
Subject	Design and Analysis of Algorithms
Regulation	R18
Subject Code	UGIT5T0118

UNIT 5

1.In analysis of algorithm, approximate relationship between the size of the job and the amount of work required to do is expressed by using __________

(a) Central tendency
(b) Differential equation
(c) Order of execution (d) Order of magnitude

2.P, Q and R are pointer variables. The statements below are intended to swap the contents of the nodes pointed to by P and Q. rewrite it so that it will work as intended.

P = Q; R = Q; Q = R;

(a) R=Q; P=R; Q=R; (b) R=P; P=P; Q=Q;
(c) P=P; P=Q; R=Q; (d) R=P; P=Q; Q=R;

3.Consider the usual algorithm for determining whether a sequence of parentheses is balanced. What is the maximum number of parentheses that will appear on the stack AT ANY ONE TIME when the algorithm analyzes: (()(())(()))

(a) 1 (b) 2 (c) 3 (d) 4

4.The Knapsack problem where the objective function is to minimize the profit is _______

(a) Greedy (b) Dynamic 0 / 1
(c) Back tracking (d) Branch & Bound 0/1

UNIT 6

Choose the correct answer for the following statements:

I. The theory of NP–completeness provides a method of obtaining a polynomial time for NPalgorithms.
II. All NP-complete problem are NP-Hard.

(a) I is FALSE and II is TRUE
(b) I is TRUE and II is FALSE
(c) Both are TRUE
(d) Both are FALSE

2.For 0/1 KNAPSACK problem, the algorithm takes_____________amount of time for memory table, and______time to determine the optimal load, for N objects and W as the capacity of KNAPSACK.

(a) O(N+W), O(NW) (b) O(NW), O(N+W)
(c) O(N), O(NW) (d) O(NW), O(N)

3.What is the type of the algorithm used in solving the 8 Queens problem?

(a)Greedy (b)Dynamic (c)Branch and Bound (d)Backtracking.

4.Sorting is not possible by using which of the following methods?

(a)Insertion (b)Selection (c)Deletion (d)Exchange

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN: BHIMAVARAM (AUTONOMOUS)
DEPARTMENT OF INFORMATION TECHNOLOGY

Objective Questions	
Department	Information Technology
Year / Semester	III B.Tech (IT) – I-Semester
Subject	Design and Analysis of Algorithms
Regulation	R18
Subject Code	UGIT5T0118

Unit-1:

(01) Which of the following properties are necessary for an Algorithm?
(A) Definite ness (B) correct ness (C) Effectiveness (D) A and C

(02) which of the following technique is not using for solve a 0-1knapsack problem
(A) Greedy (B) Dynamic programming (C) branch and bound (D) all of the above

(03)For the following program gives Big O analysis of the running time (in terms of n)
For (i=0; i<n; i++) A[i] = +;
(A) O(n-1) (B) O(n) (C) $O(n^2)$ (D) O(log n)

(04) For the following program gives Big O analysis of the running time (in terms of n)
For (i=0; i< n; i++)
For (j=i; j< n; j++)
For (k=j; k< n; k++)
S++;
(A) O(n-1) (B) $O(n^2)$ (C) $O(n^3)$ (D) O(log n)

(05) For the following program gives Big O analysis of the running time (in terms of n)
For (i=0; i < n*n; i++) A[i] = i;
(A) O(n-1) (B) $O(n^2)$ (C) $O(n^3)$ (D) O(log n)

(06) Given $f(n) = \log_2^n$, $g(n) = \sqrt{n}$ which function is asymptotically faster
(A) f(n) is faster than g(n) (B) g(n) is faster than f(n)
(C) Either f(n) or g(n) (D) Neither f(n) nor g(n)

(07) Which of the following are true
(a) $33n^3 + 4n^2 = p\ (n^2)$ (b) $n! = O(n^n)$ (c) $10n^2 + 9 = O(n^2)$ (d) $6n^3 /(\log n +1) = O(n^3)$
(A) a,b and c (B)a and c (C) a and b (D) all are true

(08) n! =
(A) $O(2^n)$ (B) $\omega(2^n)$ (C) A and B (D) $O(n^{100})$
None-

(09) T (n) = 8T (n/2) + n^2, T (1) = 1 then T (n) =
(A) $\Theta(n^2)$ (B) $\Theta(n^3)$ (C) $\Theta(n^4)$ (D) $\Theta(n)$

(10) T (n) = 3T (n/4) + n then T (n) =
(A) $O(n^2)$ (B) $O(n^3)$ (C) O (n) (D) $O(n^4)$

(11) T (n) = 4T (n/2) + n then T (n) =
(A) $\Theta(n^2)$ (B) $\Theta(n^3)$ (C) $\Theta(n^4)$ (D) $\Theta(n)$

(12) T (n) = 2T (n/2) + cn then T (n) =
(A) O (log n) (B) O (n log n) (C) $O(n^2 \log n)$ (D) $O(n^2)$

(13) T (n) = 2T (n/2) + n^2 then T (n) =
(A) $\Theta(n^2)$ (B) $\Theta(n^3)$ (C) $\Theta(n^4)$ (D) $\Theta(n)$

(14) T (n) = 2T (n/2) + n^2 then T (n) =
(A) O (n^3) (B) O (n^2) (C) O (n) (D) O (n^4)
(15) T (n) = 9T (n/3) + n then T (n) =
(A) Θ (n^4) (B) Θ (n^3) (C) Θ (n^2) (D) Θ (n)
(16) T (n) = T (n/2) + 1 then T (n) =
(A) O (log n) (B) O (2 log n) (C) O (n log n) (D) O (n2)
(17) T (n) = T (n/2) + n2 then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(18) T (n) = 4T (n/2) + n2 then T (n) =
(A) Θ (n log n) (B) Θ (n3 log n) (C) Θ (n2 log n) (D) Θ (n4 log n)
(19) T (n) = 7T (n/2) + n2 then T (n) =
(A) Θ (n2.5) (B) Θ (n2.807) (C) Θ (n2.85) (D) Θ (n2.75)
(20) T (n) = 2T (n/2) + n3 then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(21) T (n) = T (9n/10) + n then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(22) T (n) = 16T (n/4) + n2 then T (n) =
(A) Θ (n log n) (B) Θ (n3 log n) (C) Θ (n2 log n) (D) Θ (n4 log n)
(23) T (n) = 7T (n/3) + n2 then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(24) T (n) = 7T (n/2) + n2 then T (n) =
(A) Θ (nlog7) (B) Θ (nlog5) (C) Θ (nlog9) (D) Θ (nlog3)
(25) T (n) = 2T (n/2) + n3 then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(26) T (n) = 2T (n/4) + √n then T (n) =
(A) Θ (n log n) (B) Θ (√n log n) (C) Θ (n2 log n) (D) Θ (n3 log n)
(27) T (n) = T (√n) +1 then T (n) =
(A) Θ (n log n) (B) Θ (√n log n) (C) Θ (log n) (D) Θ (n2 log n)
(28) T (n) = 100T (n/99) + log (n!) then T (n) =
(A) Θ (n log n) (B) Θ (√n log n) (C) Θ (n2 log n) (D) Θ (n3 log n)
(29) T (n) = T (n-1) + n4 then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(30) T (n) = 2T (n/2) + 3n2 and T (1) = 11 then T (n) =
(A) O (n3) (B) O (n2) (C) O (n) (D) O (n4)
(31) T (n) = 1 for n=1
= 2 * T (n - 1) for n >1 then T (n) =
(A) 2 n (B) 2 n-1 (C) 2 n-2 (D) 2 n-3
(32) T (n) = 4T (n/2) + n2√n then T (n) =
(A) Θ (n3 √n) (B) Θ (n2) (C) Θ (n2√n) (D) Θ (n√n)
(33) T (n) = 2T (n/2) + (n/ log n) then T (n) =
(A) Θ (n log n) (B) Θ (n log n log n) (C) Θ (n2 log n log n) (D) Θ (n2 log n)
(34) T (n) = T (n/2) + T (n/4) + T (n/8) + n then T (n) =
(A) Θ (n4) (B) Θ (n3) (C) Θ (n2) (D) Θ (n)
(35) Set defines as
(A) Distinct objects (B) Similar elements (C) collection of elements (D) objects
(36) A machine took 200 sec to sort 200 names, using bubble sort. In 800 sec, it can approximately sort
(A) 400 names (B) 800 names (C) 750 names (D) 1800 names
(37) Linked lists are not suitable for
(A) Insertion sort (B) Binary search (C) Radix sort (D) Polynomial manipulation

(38) Which of the following is useful in implementing quick sort?
(A) Stack (B) List (B) Set (D) Queue
(39) A machine needs a minimum of 100 sec to sort 1000 names by quick sort. The minimum time needed to sort 1000 names by quick sort. The minimum time needed to sort 100 names will be approximately?
(A) 50.2 sec (B) 6.7 sec (C) 72.7 sec (D) 11.2 sec
(40) Given 2 sorted lists of size 'm' and 'n' respectively. Number of comparisons needed in the worst case by the merge sort algorithm will be
(A) mn (B) max(m,n) (C) min(m,n) (D) m+n-1
(41) The depth of a complete binary tree with 'n' nodes is
(A) log (n+1)-1 (B) log n (C) log (n-1)+ 1 (D) log n +1
(42) Average successful search time taken by binary search on a sorted array of items is
(A) 2.6 (B) 2.7 (C) 2.8 (D) 2.9
(43) Average successful search time for sequential search on 'n' items is
(A) n/2 (B) (n-1)/2 (C) (n+1)/2 (D) n2
(44) The maximum number of comparisons needed to sort 7 items using radix sort is (assume each item is a 4 digit decimal number)
(A) 280 (B) 40 (C) 47 (D) 38
(45) In Randomized Quick sort, the expected running time of any input is
(A) O(n) (B) O(n2) (C) O(n log n) (D) O(n3)
(46) If Total complexity after micro analysis is $5n^3 + 10n^2 + 100\,n + 400\log n + 10$,
The Big Oh complexity is
(A) $O(n^2)$ (B) $O(n^3)$ (C) O(nlogn) (D) $O(n^2 \log n)$
(47) In Strassen's Multiplication Algorithm the T(n) is
A) $7T(n) + bn^2$ B) $7T(n/2) + bn^2$ C) $8T(n/2) + bn^2$ D) 7T (n/2) + bn
(48) T (n) = 4 T (n/2) + n then in Big Oh Notation it is
(A) $O(n^2)$ B) O(4) (C) O(n) D) O(log(n))
(49) In T(n) = a * T(n/b) + f(n) , a refers to
(A) Size of sub problem (B) No. of sub problems
(C) Size of the problem (D) Time to combine solutions
(50) 0-1 knapsack be solved using
(A) dynamic programming (B) Backtracking (C) Branch & Bound
(D) All A,B,C,E (E) Genetic Programming
(51) In depth first search algorithm the no. of recursive calls we have to make are
(A) 2 (B) 1 (C) 6 (D) depends on the graph
(52) O (f(n)) minus O(f(n)) is equal to
(A) Zero (B) A constant (C) f(n) (D) O(f(n))
(53) Quick sort is solved using
(A) Divide and conquer (B) Greedy Programming
(C) Dynamic Programming (D) Branch and bound
(54) For i = 1 to n-1 do
1.1 For j = 1 to n-1-i do
2.2.1 If (a[j+1] < a[j]) then swap a[j] and a[j+1]
Given code is for
(A) Bubble sort (B) Insertion sort (C) Quick Sort (D) Selection Sort
(55) Worst case complexity of quick sort is
(A) O(n) (B) O(logn) (C) O(nlogn) (D) $O(n^2)$
(56) The sub problems in Divide and Conquer are considered to be
A) Distinct (B) overlapping (C) large size (D) small size

(57) Which of the following name does not relate to stacks?
(A) FIFO lists (B) LIFO list (C) Piles (D) Push-down lists
(58) Which of the following data structure is linear type?
(A) Strings (B) Lists (C) Queues D) All of above
(59) In a graph if e=(u, v) means
(A) u is adjacent to v but v is not adjacent to u (B) e begins at u and ends at v
(C) u is processor and v is successor (D) both b and c
(60) An algorithm that calls itself directly or indirectly is known as
(A) Sub algorithm (B) Recursion (C) Polish notation (D) Traversal algorithm
(61) In a Heap tree
(A) Values in a node is greater than every value in left sub tree and smaller than right sub tree
(B) Values in a node is greater than every value in children of it
(C) Both of above conditions applies (D) none of above conditions applies
(62) The postfix form of the expression (A+ B)*(C*D− E)*F / G is
(A) AB+ CD*E − FG /** (B) AB + CD* E − F **G /
(C) AB + CD* E − *F *G / (D) AB + CDE * − * F *G /
(63) What is the postfix form of the following prefix expression -A/B*C$DE
(A) ABCDE$*/- (B) A-BCDE$*/-
(C) ABC$ED*/- (D) A-BCDE$*/
(64) You have to sort a list L consisting of a sorted list followed by a few “random” elements. Which of the following sorting methods would be especially suitable for such a task?
(A) Bubble sort (B) Selection sort
(C) Quick sort (D) Insertion sort
(65) A technique for direct search is
(A) Binary Search (B) Linear Search
(C) Tree Search (D) Hashing
(66) The searching technique that takes O (1) time to find a data is
(A) Linear Search (B) Binary Search (C) Hashing (D) Tree Search
(67) A mathematical-model with a collection of operations defined on that model is called
(A) Data Structure (B) Abstract Data Type
(C) Primitive Data Type (D) Algorithm
(68) The complexity of multiplying two matrices of order m*n and n*p is
(A) mnp (B) mp (C) mn (D) np
(69) In worst case Quick Sort has order
(A) O (n log n) (B) O (n2/2) (C) O (log n) (D) O (n2/4)
(70) A full binary tree with n leaves contains
(A) n nodes. (B) log 2 n nodes. (C) 2n –1 nodes. (D) n 2 nodes.
(71) The quick sort algorithm exploit__________design technique
(A) Greedy (B) Dynamic programming (C)Divide and Conquer (D)Backtracking
(72) The maximum degree of any vertex in a simple graph with n vertices is
(A) n–1 (B) n+1 (C) 2n–1 (D) n
(73) The total number of companions required to merge 4 sorted files containing 15, 3, 9 and 8 records into a single sorted file is
(A) 66 (B) 39 (C) 15 (D) 33
(74) The number of leaf nodes in a complete binary tree of depth d is
(A) 2d (B) 2d-1 +1 (C) 2d+1 +1 (D)2d +1

(75) If x is initialize as x=100. What will be the value of x and y after step-4?
Step 1 x=100;
Step 2 Y=x++;
Step 3 x=x+y ;
Step 4 Y=++x;
(a)302,201 (b) 201,302 (c)101 ,100 (d)None of these

(76) Struct x
```
{
int i;
char c;
}
 union  y{
struct x a;
double d;
};
 printf("%d",sizeof(union y));
```
(A)8 (B)5 (C)4 (D)1

(77) Worst case complexity of the insertion sort algorithm is
(A) O(n2) (B) O(n) (C) O(n-1) (D) O(n+1)

(78) Average case complexity of the insertion sort algorithm is
(A) O(n2) (B) O(n) (C) O(n-1) (D) O(n+1)

(79) Best case complexity of the insertion sort algorithm is
(A) O(n2) (B) O(n) (C) O(n-1) (D) O(n+1)

(80) Worst case complexity of the bubble sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(81) Best case complexity of the bubble sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(82) Average case complexity of the bubble sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(83) Worst case complexity of the selection sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(84) Average case complexity of the selection sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(85) Best case complexity of the selection sort algorithm is
(A) O(n3) (B) O(n4) (C) O(n2) (D) O(n)

(86) If a complete binary tree Tn has n=1000 nodes then its height is
(A) 21 (B) 10 (C) 11 (D) 12

(87) If a complete binary tree Tnhas n=1000000 nodes then its height is
(A) 21 (B) 20 (C) 23 (D) 22

(88) The running time of Strassen's algorithm for matrix multiplication is
(A) ϴ (n) (B) ϴ (n3) (C) ϴ (n2) (D) ϴ (n2.81)

(89) The running time of Floyd-Warshall algorithm is
(A) ϴ (n) (B) ϴ (n3) (C) ϴ (n2) (D) ϴ (n log n)

(90) Dijkastra's algorithm bears some similarity to
(A) BFS (B) prim's algorithm (C) DFS (D) Both (A) & (C)

(91) The running time of Dijkastra's algorithm is
(A) $O(V^2)$ (B) O(V+E) (C) O(n log n) (D) all of the above

(92) kruskal's algorithm uses-------- and prim's algorithm uses------ in determining the MST
(A) edges,vertex (B)vertex,edges (C)edges,edges (D)vertex,vertex

(93) The running time of kruskal's algorithm for MST
(A) O(E) (B) O(V) (C) O(E log V) (D) all of the above
(94) We can perform a topological sort in time -----------, since DFS takes -------time.
(A) Θ (V+E), Θ (E) (B) Θ (E), Θ (V+E)
(C) Θ (V+E), Θ (V+E) (D) all of the above
(95) The running time of BFS is------------
(A) Θ (1) (B) Θ (n log n) (C) O(V+E) (D) Θ (n^2)
(96) For ----- insertion sort beats merge sort
(A) n≥43 (B) n≤23 (C) n≤43 (D) cannot say
(97) Best case running time of quick sort is
(A) O(n) (B) O(logn) (C) O(nlogn) (D) O(n^2)
(98) A characteristic of the data that binary search tree but the linear search ignores, is the
(A) Order of the list (B) length of the list
(C) maximum value in the list (D) mean of data values
(99) A sort which compares adjacent elements in a list and switches where necessary is a
(A) insertion sort (B) heap sort
(C) quick sort (D) bubble sort
(100) A sort which iteratively passes through a list to exchange the first element with any element less than it and then repeats with a new first element is called
(A) Insertion sort (B) selection sort
(C) Heap sort (D) quick sort
(101) A sort which uses the binary tree concept such that any number is larger than all the numbers is the subtree below it is called
(A) Selection sort (B) insertion sort (C) quick sort (D) heap sort
(102) which of the sorting algorithm does not have a worst case running time of O(n^2)
(A) Selection sort (B) insertion sort (C) merge sort (D) quick sort
(103) which of the following sorting method is stable?
(A) Straight insertion sort (B) binary search tree
(C) Shell sort (D) Heap sort
(104) A complete binary tree with the property that the value at each node is at least as large as the values at its children is known as
(A) Binary search tree (B) AVL tree
(C) Completely balanced tree (D) Heap
(105) The recurrence relation T(n) = mT(n/2)+ an^2 is satisfied by
(A) T (n) = O(n^m) (B) T(n) = O(n log m) (C) T(n) = O(n log n) (D) T(n) =O(m log n)
(106)The time required to find shortest path in a graph with n vertices and e edges is
(A) O (e) (B) O (n) (C) O (e^2) (D) O (n^2)
(107)The goal of hashing is to produce a search tree that takes
(A) O(1) time (B) O(n^2) time (C) O (log n) time (D) O (n log n) time
(108) which of the following best described sorting?
(A) Accessing and processing each record exactly once
(B) Finding the location of the record with a given key
(C) Arranging the data in some given order
(D) Adding a new record to the data structure
(109) The worst case complexity of straight insertion sort algorithm to sort n elements is
(A) O(n) (B) O(n log n) (C) O($n^{1.2}$) (D) O(n^2)

(110) The worst case complexity of binary insertion sort algorithm to sort in n elements is
(A) O(n) (B) O(n log n) (C) $O(n^{1.2})$ (D) $O(n^2)$
111) If each node in a tree has value greater than every value in its left sub tree and value less than every value in its right sub tree, the tree is known as
(A) complete tree (B) full binary tree
(C) binary search tree (D) threaded tree
(112) Which of the following sorting procedure is the slowest?
(A) Quick sort (B) Heap sort (C) Shell sort (D) Bubble sort
(113) which of the following shows the correct relationship among some of the more common computing times on algorithms
(A) $O(\log n) < O(n) < O(n^* \log n) < O(2^n) < O(n^2)$
(B) $O(n) < O(\log n) < O(n^* \log n) < O(2^n) < O(n^2)$
(C) $O(n) < O(\log n) < O(n^* \log n) < O(n^2) < O(2^n)$
(D) $O(\log n) < O(n) < O(n^* \log n) < O(n^2) < O(2^n)$
(114) The average time required to perform a successful sequential search for an element in an array A(1..n) is given by
(A) (n+1)/2 (B) n(n+1)/2 (C) log n (D) n^2
(115) the time complexity of linear search algorithm over an array of n elements is
(A) O(log n) (B) O(n) (C) O(n log n) (D) $O(n^2)$
(116) the time taken by binary search algorithm to search a key in a sorted array of n elements is
(A) O(log n) (B) O(n) (C) O(n log n) (D) $O(n^2)$
(117) the time required to search an element in a linked list of length n is
(A) O(log n) (B) O(n) (C) O(1) (D) $O(n^2)$
(118) the worst case time required to search a given element in sorted linked list of length n is
(A) O(1) (B) O(log n) (C) O(n) (D) O(n log n)
(119) consider a linked list of n elements which is pointed by an external pointer. What is the time taken to delete the element which is successor of the element pointed to by a given pointer?
(A) O(1) (B) O(log n) (C) O(n) (D) O(n log n)
(120) consider a linked list of n elements. What is the time taken to insert an element an after element pointed by some pointer?
(A) O(1) (B) O(log n) (C) O(n) (D) O(n log n)
(121) which of the following operations is performed more efficiently by doubly linked list than by linear linked list?
(A) Deleting a node whose location is given
(B) searching an unsorted list for a given item
(C) inserting a node after the node with a given location
(D) Traversing the list to process each node
(122) the five items: A,B,C,D and E are pushed in a stack, one after the other starting from A. The stack is popped four items and each element is inserted in a queue. Then two elements are deleted from the queue and pushed back on the stack. Now on item is popped from the stack. The popped item is
(A) A (B) B (C) C (D) D
(123) the time required to search an element in a binary search tree having n elements is
(A) O(1) (B) O(log n) (C) O(n) (D) O(n log n)
(124) for a linear search in an array of n elements the time complexity for best, worst and average case are …., and …respectively.
(A) O(n) , O(1) and O(n/2) (B) O(1) , O(n) and O(n/2)
(C)) O(1) , O(n) and O(n) (D)) O(1) , O(n) and O((n-1)/2)

(125) the number of comparisons required by binary search of 100000 elements is
(A) 15 (B) 20 (C) 25 (D) 30
(126) Find an optimal parenthesization of a matrix chain product whose sequence of dimension s is <5,4,6,2,7>
(A) 156 (B) 154 (C) 158 (D) 157
(127) Find an optimal parenthesization of a matrix chain product whose sequence of dimension s is <5,10,3,12, 5, 50, 6>
(A) 2010 (B) 2020 (C) 2015 (D) 2030
(128) Find an optimal parenthesization of a matrix chain product whose sequence of dimension s is <4,10,3,12,20,7>
(A) 1334 (B) 1324 (C)1344 (D)1354
(129) Find an optimal parenthesization of a matrix chain product whose sequence of dimension s is <5,4,3> (for three matrices)
(A) 125 (B) 130 (C) 135 (D) 140
(130) Find an optimal parenthesization of a matrix chain product whose sequence of dimension s is <30,35,15,5,10,20,25> (for six matrices)
(A)7130 (B) 7125 (C) 7145 (D) 7135
(131) there are 5 items as follows

Items	w_i	v_i
Item1	5 pounds	30$
Item2	10 pounds	20$
Item3	20 pounds	100$
Item4	30 pounds	90$
Item5	40 pounds	160$

The knapsack can hold 60 pounds find the optimal solution
(A) 250$ (B) 260 $ (C) 270 $ (D) 290$
(132) there are 5 items as follows

Items	w_i	v_i
Item1	5 pounds	30$
Item2	10 pounds	20$
Item3	20 pounds	100$
Item4	30 pounds	90$
Item5	40 pounds	160$

The knapsack can hold 60 pounds find the solution by greedy technique

(A) 230$ (B) 260 $ (C) 220 $ (D) 250$
(133) what is an optimal Huffman code for alphabeta of the following set of frequencies a: 05, b:48, c:07, d:17, e:10, f:13
(A) 1010 (B)0101 (C) 1001 (D) 1100
(134) the total running time of Huffman on the set of n characters is
(A) O(n) (B) O(n log n) (C) $O(n^2)$ (D) O(log n)
(135) the total running time of matrix chain multiplication of n matrices
(A) $\Theta(n^4)$ (B) $\Theta(n^3)$ (C) $\Theta(n^2)$ (D) $\Theta(n)$
(136) which of the following is true
(A) P is subset of NP (B) NP is subset of P
(C) P and NP are equal (D) NP is subset of NP hard
(137) the total running time of optimal binary search tree of n nodes
(A) $O(n^2)$ (B) O(n) (C) $O(n^3)$ (D) O(n log n)

(138) If every square of the board is visited, then the total number of knight moves of n-queen problem is
(A) n^3-1 (B) n-1 (C)n^2-1 (D) log n-1
(139) If every square of the board is visited, then the total number of knight moves of 4-queen problem is
(A) 14 (B) 15 (C) 16 (D) 12
(140) If every square of the board is visited, then the total number of knight moves of 8-queen problem is
(A) 64 (B) 62 (C) 61 (D) 63
(141) In which of the following cases n-queen problem does not exist
(A) n=2 and n=4 (B) n=4 and n=6 (C) n=2 and n=3 (D) n=4 and n=8
(142) the total running time of knapsack problem for a simple approach
(A) O(n) (B) O(log n) (C) O(2^n log n) (D) O(2^n)
(143) what is an optimal Huffman code for alphabeta of the following set of frequencies a: 01, b:01, c:02, d:03, e:05, f:8, g:13, h:21
(A) 001010 (B) 001111 (C) 111100 (D) 101010
(144) what is an optimal Huffman code for alphabet b of the following set of frequencies a: 45, b:13, c:12, d:16, e:9, f:5
(A) 100 (B) 111 (C) 001 (D) 101
(145) what is an optimal Huffman code for alphabete of the following set of frequencies a: 29, b:25, c:20, d:12, e:05, f:09
(A) 100 0 (B) 1110 (C) 0010 (D) 1011
(146) Which of the following method is taking overcharge for some operations in amortized analysis?
(A) Aggregate method (B) accounting method
(C) potential method (D) both (A) and (C)
(147) Which of the following method is most flexible in amortized analysis?
(A) Aggregate method (B) accounting method
(C) potential method (D) both (A) and (B)
(148) Which of the following method is taken different operations different charges in amortized analysis?
(A) Aggregate method (B) accounting method
(C) potential method (D) both (A) and (B)
(149) Which of the following method is computing total cost of an algorithm in amortized analysis?
(A) Aggregate method (B) accounting method
(C) potential method (D) both (C) and (B)
(150) which of the following method is credit as the potential energy to pay for future operations?
(A) Aggregate method (B) accounting method
(C) potential method (D) both (A) and (B)
(151) If all c(i, j)'s and r(i, j)'s are calculated, then OBST algorithm in worst case takes one of the following time.
(a) O(n log n)
(b) O(n^3)
(c) O(n^2)
(d) O(log n)
(e) O(n^4).

(152) The following is a weighted binary tree, then what is the weighted array for the TVS problem?
(a) [9, 2, 7, 0, 0, 0, 0, 0, 0, 0, 0, 0, 6, 4]
(b) [9, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 4, 6]
(c) [9, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 6, 7, 4]
(d) [9, 2, 0, 0, 0, 7, 0, 0, 0, 0, 0, 0, 6, 4]
(e) [9, 2, 0, 0, 0, 7, 0, 0, 0, 0, 6, 4, 0, 0]
(153) The upper bound on the time complexity of the nondeterministic sorting algorithm is
(a) O(n)
(b) O(n log n)
(c) O(1)
(d) O(log n)
(154) The worst case time complexity of the nondeterministic dynamic knapsack algorithm is
(a) O(n log n)
(b) O(log n)
(c) O(n^2)
(d) O(n)
(155) The time complexity of the normal quick sort, randomized quick sort algorithms in the worst case is
(a) O(n^2), O(n log n) (b) O(n^2), O(n^2)
(c) O(n log n), O(n^2) (d) O(n log n), O(n log n)
(156) Let there be an array of length 'N', and the selection sort algorithm is used to sort it, how many times a swap function is called to complete the execution?
(a) N log N times (b) log N times
(c) N2 times (d) N-1 times
(157) The Sorting method which is used for external sort is
(a) Bubble sort (b) Quick sort (c) Merge sort (d) Radix sort
(158) In analysis of algorithm, approximate relationship between the size of the job and the amount of work required to do is expressed by using __________
(d) Central tendency
(e) Differential equation
(f) Order of execution (d) Order of magnitude
(159) P, Q and R are pointer variables. The statements below are intended to swap the contents of the nodes pointed to by P and Q. rewrite it so that it will work as intended.
P = Q; R = Q; Q = R;
(a) R=Q; P=R; Q=R; (b) R=P; P=P; Q=Q;
(c) P=P; P=Q; R=Q; (d) R=P; P=Q; Q=R;
(160) Consider the usual algorithm for determining whether a sequence of parentheses is balanced. What is the maximum number of parentheses that will appear on the stack AT ANY ONE TIME when the algorithm analyzes: (()(())(()))
(a) 1 (b) 2 (c) 3 (d) 4
(161) The Knapsack problem where the objective function is to minimize the profit is ______
(a) Greedy (b) Dynamic 0 / 1
(c) Back tracking (d) Branch & Bound 0/1

(162) Choose the correct answer for the following statements:
III. The theory of NP–completeness provides a method of obtaining a polynomial time for NPalgorithms.
IV. All NP-complete problem are NP-Hard.
(e) I is FALSE and II is TRUE
(f) I is TRUE and II is FALSE
(g) Both are TRUE
(h) Both are FALSE
(163) For 0/1 KNAPSACK problem, the algorithm takes_________amount of time for memory table, and______time to determine the optimal load, for N objects and W as the capacity of KNAPSACK.
(a) O(N+W), O(NW) (b) O(NW), O(N+W)
(c) O(N), O(NW) (d) O(NW), O(N)
(164) What is the type of the algorithm used in solving the 8 Queens problem?
(a)Greedy
(b)Dynamic
(c)Branch and Bound
(d)Backtracking.
(165) Sorting is not possible by using which of the following methods?
(a)Insertion
(b)Selection
(c)Deletion
(d)Exchange

Sub Code: UGIT5T0118

SHRI VISHNU ENGINEERING COLLEGE FOR WOMEN:: BHIMAVARAM
(AUTONOMOUS)
(AICTE Approved &Permanently Affiliated to JNTUK, Kakinada)
DESIGN AND ANALYSIS OF ALGORITHMS QUESTION PAPER

Examination: Year: III Semester: I

Time: 3hrs Max.Marks:60

Answering one Question from each unit,
All Questions carry equal marks.

UNIT-I

1. What are the different techniques to represent an algorithm. Explain? (10M)

OR

2. Give an algorithm to solve the towers of Hanoi problem. (10M)

UNIT-2

3. Write the merge sort algorithm and Draw the tree of calls of merge for the following set of elements (10M)
(20, 30, 10, 40, 5, 60, 90, 45, 35, 25, 15, 55)

OR

4. Write an algorithm for quick sort by using recursive method. (10M)

UNIT-3

5.Find the solution for the knapsack problem. When n=3, (10M)
(W_1, W_2, W_3)=(18, 15, 10,) . (P1, P2, P3)= (25, 24, 15) and m=20.

OR

6.Explain the general concept of Dynamic programming. (10M)

UNIT-4

7. Explain, how the Hamiltonian circuit problem is solved by using the backtracking concept. (10M)

OR

8.Device a backtracking algorithm for m-coloring graph problem. (10M)

UNIT-5

9.Explain how the traveling salesperson problem is solved by using LC Branch and Bound. .

OR

10.Write the general algorithm for Branch and Bound. . (10M)

UNIT-6

11 . Explain the 0/1 knapsack problem. Consider the following instance of the knapsack problem n=3,m=20,
(p1,p2,p3)=(25,24,15), and (w1,w2,w3)=(18,15,10). (10M)

12. Explain the NP-hard and NP-complete problems (10M)