



Date: 30-04-2025

Dept. No.

Max. : 100 Marks

Time: 09:00 AM - 12:00 PM

SECTION A – K1 (CO1)

Answer ALL the questions

(5 x 1 = 5)

1 Answer the following

a) Write the output of the following code when n = 4.

```
def compute(n):
    j=1
    for i in range(1,n):
        while j<= n:
            j=j*2
            print(j)
```

b) Identify the linked list data structure which doesn't contain NULL pointer in any of the nodes.

c) Draw the tree of calls for Algorithm MergeSort when n = 7.

d) Distinguish between greedy and dynamic programming strategies.

e) State Cook's theorem.

SECTION A – K2 (CO1)

Answer ALL the questions

(5 x 1 = 5)

2 Multiple Choice Questions

a) The output of the given code is

```
a = "DATA STRUCTURES"
print(a.find("T"))
```

(i) 1 (ii) 2 (iii) 3 (iv) 6

b) Consider a queue data structure of size 8. If front = 2 and rear = 7, then the next element is inserted at

(i) q[0] (ii) q[1] (iii) q[7] (iv) q[8].

c) The best-case time complexity of Algorithm BinSearch is

(i) O(1) (ii) O(n) (iii) O(logn) (iv) O(nlogn)

d) The optimal ordering of tapes with length 15, 6, 25, 31, 42 is

(i) 6, 15, 25, 31, 42.
 (ii) 15, 31, 6, 15, 42.
 (iii) 42, 31, 25, 15, 6.
 (iv) 31, 6, 15, 25, 42.

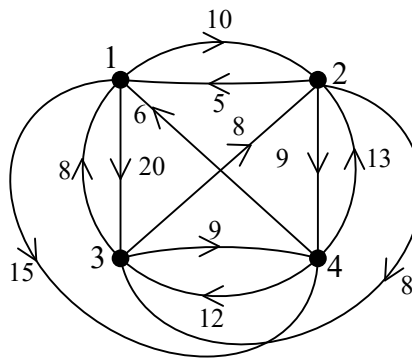
- e) In backtracking strategy, a constraint is
- (i) a condition that a solution satisfies.
 - (ii) a decision point.
 - (iii) all possible solutions.
 - (iv) a criterion function.

SECTION B – K3 (CO2)

Answer any **THREE** of the following

(3 x 10 = 30)

- 3 Design an algorithm to compute the sum $\sum_{i=1}^n \frac{i^2+2i+1}{n}$ and present its relevant code.
- 4 How would you insert and delete element from a doubly linked-list? Develop a code to perform the same.
- 5 Formulate an algorithm which partitions the array $a[m : p - 1]$ about $a[m]$. Execute the algorithm on the array $a[1 : 7] = (50, 70, 80, 60, 15, 4, 20)$.
- 6 Propose an algorithm to solve travelling salesman problem and find the optimal tour starting from the vertex labelled 1 for the following graph:



- 7 What are nondeterministic algorithms and give a nondeterministic sorting algorithm.

SECTION C – K4 (CO3)

Answer any **TWO** of the following

(2 x 12.5 = 25)

- 8 Formulate an algorithm to insert and delete an element from a circular queue data structure. If the circular queue is represented by $q[0 : 6] = (2, 6)$, then what is the output after inserting the elements 10, 8 into $q[0 : 6]$?
- 9 Design a recursive sorting algorithm which merges two sorted sets. Simulate it on the array $a(1 : 6) = (85, 65, 15, 5, 95, 25)$.
- 10 Present an algorithm to find a minimum cost spanning tree. Use the algorithm to find the minimum cost spanning tree for a graph having 7 vertices, 12 edges. Assign weights 5, 6, 7 to any of the three edges and also assign the weight 3 for rest of the edges.
- 11 Outline a backtracking algorithm to find all possible subsets of the set w that sums to m and give its implementation in Python. What will be the output when $w = \{1, 2, 3, 5, 7, 10\}$ and $m = 13$.

SECTION D – K5 (CO4)

Answer any ONE of the following

(1 x 15 = 15)

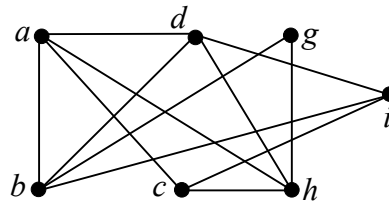
- 12 Develop a recursive algorithm to search an element with key x in a binary search tree and provide its implementation in Python. Validate the algorithm on a binary search tree with key values 55, 65, 14, 30, 20, 60, 7, 10, 57 and $x = 30, 56$.
- 13 Formulate an algorithm which generates a two-way merge tree and establish its proof of correctness. Obtain the optimal binary merge pattern for eight files with length 13, 18, 15, 9, 4, 2, 7, 8 and construct a binary merge tree representing the merge pattern.

SECTION E – K6 (CO5)

Answer any ONE of the following

(1 x 20 = 20)

- 14 State the algorithm Heapify and compute its worst-case complexity.
Design an algorithm which uses algorithm Heapify for sorting an array and illustrate its performance on the sequence of the numbers 56, 3, 99, 45, 21, 80, 14.
- 15 For a connected graph G , devise an algorithm for breadth first search traversal. If the start vertex is b , then trace a spanning tree for the following graph using the algorithm. Also, give its implementation in Python.



\$\$\$\$\$\$\$\$\$\$\$\$