LOYOLA COLLEGE (AUTONOMOUS), CHENNAI – 600 034

M.Sc. DEGREE EXAMINATION – **MATHEMATICS**

FIRST SEMESTER – **NOVEMBER 2022**

PMT1MC04 – DATA STRUCTURES AND ALGORITHMS USING PYTHON

Date: 30-11-2022 Dept. No. Max. : 100 Marks

Time: 01:00 PM - 04:00 PM

SECTION - A

Answer ALL the Questions:

1.	**Answer the following:**	**(5 x 1 = 5)**	
a)	Identify the difference between break and continue statements in Python.	K1	CO1
b)	List any two applications of the data structure queue.	K1	CO1
c)	Define the performance measures of an algorithm.	K1	CO1
d)	Write the principle of optimality.	K1	CO1
e)	State Cook's theorem.	K1	CO1
2.	**Choose the correct answer:**	**(5 x 1 = 5)**	
a)	How many times the following program segment will be executed? *i* = **input**('Enter the number') **while** *i* != 0: **print**(2 ** *i*) (i) 0 (ii) 1 (iii) 2 (iv) infinite	K2	CO1
b)	Which of the following data structure is used to represent the linear relationship between elements by means of sequential memory locations? (i) Tree (ii) Array (iii) Stack (iv) Queue	K2	CO1

c)	The frequency count of all statements in the following algorithm segment is
	for $i \leftarrow 0$ **to** $n-1$ **do**
	$b \leftarrow 0$
	for $j \leftarrow 0$ **to** $n-1$ **do**
	$b \leftarrow b + x[i]$
	$A[i] \leftarrow b * (i+1)$
	return A
	(i) n^2+n+1 (ii) n^2+5n+2 (iii) n^2+n+2 (iv) n^2+n+3
		K2	CO1

d)	Let *G* be a connected graph with 16 vertices and 25 edges. The weight of a minimum spanning tree is 60. If the weight of each edge of *G* is increased by 5, then the weight of a minimum spanning tree is (i) 60 (ii) 120 (iii) 135 (iv) 225	K2	CO1
e)	The following is the list of nodes of a tree *T* given in sequential order: <table><tr><td>A</td><td>B</td><td>C</td><td>D</td><td>--</td><td>E</td></tr></table> Which of the following is the postorder traversal of the tree? (i) ABCDE (ii) DBACE (iii) ABDCE (iv) DBECA	K2	CO1

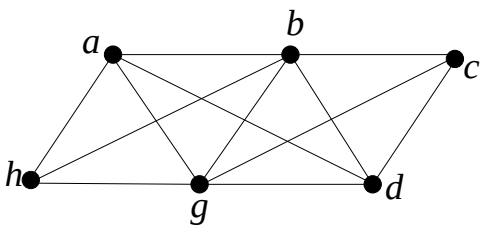
SECTION - B

Answer any THREE Questions: **(3 x 10 = 30)**

3.	Explain the need of using conditional branching statements. Illustrate with suitable Python code.	K3	CO2
4.	Write a Python function to test a given string is a palindrome using deque.	K3	CO2
5.	State Algorithm Partition and illustrate its execution on the array A[1 : 8] = (23, 13, 45, 51, 4, 60, 15, 21).	K3	CO2
6.	Formulate an algorithm for optimal storage on Tapes using greedy strategy and simulate on three tapes T_0, T_1, T_2 and programs of lengths 12, 5, 56, 34, 22, 44, 88, 66, 45, 9.	K3	CO2
7.	Explain satisfiability problem and present an algorithm to determine whether a propositional formula is satisfiable.	K3	CO2

SECTION - C

Answer any TWO Questions: **(2 x 12.5 = 25)**

8.	Develop a Python code to perform insertion and deletion operations on a stack. Show how the code works when there are 6 insertions and 5 deletions on a stack which is initially empty.	K4	CO3
9.	Form a heap from the array A[1 : 7] = (56, 13, 5, 23, 61, 40, 33) using Algorithm Heapify.	K4	CO3
10.	Design an algorithm to solve the longest common subsequence problem using dynamic programming. Use it to find the longest subsequence in the strings 'ABCBEAD' and 'BCEADC'.	K4	CO3
11.	Construct a breadth first search tree with start vertex *a* for the following graph: 	K4	CO3

	SECTION D		
Answer any ONE Question:			**(1 x 15 = 15)**
12.	Give a Python implementation to create and print the elements of a binary search tree. Run the code for the input list 60, 25, 75, 15, 33, 14.	K5	CO4
13.	Present an algorithm using greedy strategy to obtain the optimal solution for knapsack problem. Prove that the algorithm generates an optimal solution and determine the optimal solution to the instance: $n = 4$, $m = 13$, $(p_1, p_2, p_3, p_4) = (40, 42, 25, 12)$ and $(w_1, w_2, w_3, w_4) = (4, 7, 5, 3)$.	K5	CO4
	SECTION E		
Answer any ONE Question:			**(1 x 20 = 20)**
14.	Create a recursive sort algorithm which merges two sorted arrays using divide and conquer strategy and find the worst-case time complexity of the algorithm. Run the algorithm on the inputs 77, 27, 67, 37, 47, 7, 57, 17 and trace the tree of calls.	K6	CO5
15.	Develop an algorithm to determine all possible subsets of a set w that sums to m using backtracking technique. Run the algorithm when $w = \{2, 6, 8, 10, 12\}$ and $m = 20$. Also, draw the portion of state space tree generated by SumOfSub. Propose a real-time problem which can be solved using backtracking technique.	K6	CO5

@@@@@@