U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

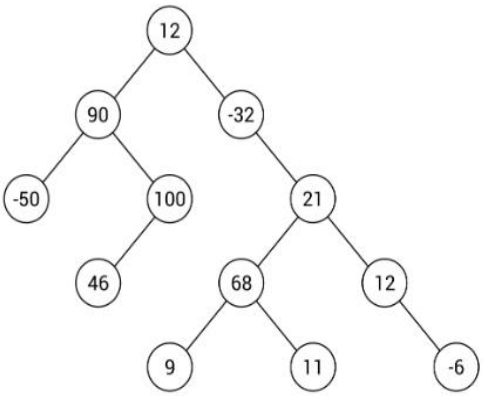
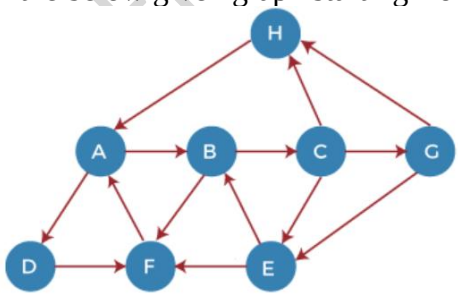
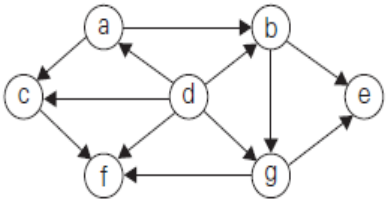
October 2024 Supplementary Examinations

Programme: B.E. **Semester: VI**
Branch: Institutional Elective **Duration: 3 hrs.**
Course Code: 22IS6OEDSA **Max Marks: 100**
Course: Data Structures and Algorithms

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	CO	PO	Marks
1	a)	Recommend a menu driven Program in C to have employee database of BMS College of Engineering for the following operations on Doubly Linked List of Employee Data with the fields: EID, Name, Department, Designation, Salary, and Phone Number. i. Create a DLL using end Insertion ii. Perform Insertion and Deletion at the end of DLL iii. Perform Insertion and Deletion at the front of DLL	*CO2*	*PO 1,3*	**08**
	b)	Discuss the working principle of a Singly Linked List. Write a C program to demonstrate how a SLL works.	*CO2*	*PO1,3*	**08**
	c)	Describe the process of dynamic memory allocation using malloc() in C. Include the syntax, its return type, and how memory is allocated in the heap. Provide a code example demonstrating its usage.	*CO2*	*PO1,3*	**04**
		UNIT - II			
2	a)	i) Convert the following infix expression to postfix expression **(a+b)*c−(d+e)÷f*(g+h)−i** ii) Evaluate the following Postfix Expressions: **abc*+de*f+g*+ where a=1, b=2, c=3, d=4, e=5, f=6, g=2**	*CO2*	*PO1,3*	**08**
	b)	How can dynamic memory allocation be used in the context of stack operations? Develop a C program for the same. Provide a scenario where dynamic memory is beneficial for stack management.	*CO2*	*PO1,3*	**06**
	c)	Consider a Circular Queue of size 05 wherein front=2 and rear = 4. Find the position of front and rear after the following operations. Enqueue, Enqueue, Dequeue, Enqueue, Dequeue, Enqueue, Dequeue, Dequeue, Enqueue.	*CO2*	*PO1,3*	**06**
		OR			
3	a)	Evaluate the following Postfix Expressions: **623+-382/+*2^3+** **abc*+de*f+g*+ where a=1, b=2, c=3, d=4, e=5, f=6, g=2.**	*CO2*	*PO1,3*	**08**

		b)	In printers, a linear queue is often employed to manage print jobs. Each print job is added to the end of the queue, and the printer processes jobs in the order they are received. Write a C program to simulate the working of the same.	*CO2*	*PO1,3*	**08**
		c)	Write C function for each of the following i. Factorial of a given number. ii. Given 'n' Fibonacci numbers.	*CO2*	*PO1,3*	**04**
			UNIT - III			
	4	a)	Give the array and linked list representation the following tree in the memory. Also perform all the tree traversals. 	*CO1*	*PO1,3*	**10**
		b)	Give the general plan for designing a recursive algorithm. Write a recursive algorithm to calculate the factorial of a number and FIBO(n).	*CO1*	*PO1,3*	**10**
			UNIT - IV			
	5	a)	Sort the given array using merge sort strategy. arr = [12, 45, 23, 6, 78, 34, 90, 15, 67, 29] Give the algorithm	*CO3*	*PO1,3*	**08**
		b)	Apply BFS for the below given graph starting from vertex H 	*CO3*	*PO1,3*	**08**
		c)	Apply bubble sort on the give list of elements. 12,88,1,33,99,2,8,65.	*CO3*	*PO1,3*	**04**
			OR			
	6	a)	Starting at vertex '**a**' and resolving ties by the vertex alphabetical order, traverse the graph by depth-first search and construct the corresponding DFS tree and BFS tree. 	*CO3*	*PO1,3*	**12**

		b)	Apply Selection sort on the give list of elements. 53,10,2,19,85,11,8,3,32,17,99,25	*CO3*	*PO1,3*	**08**
			UNIT - V			
	7	a)	Solve the all-pairs shortest-path problem for the digraph with the following weight matrix: $\begin{bmatrix} 0 & 2 & \infty & 1 & 8 \\ 6 & 0 & 3 & 2 & \infty \\ \infty & \infty & 0 & 4 & \infty \\ \infty & \infty & 2 & 0 & 3 \\ 3 & \infty & \infty & \infty & 0 \end{bmatrix}$	*CO3*	*PO1,3*	**08**
		b)	For the input 30, 20, 56, 75, 31, 19 and hash function h(K) = K mod 11 a. Construct the open hash table. b. Find the largest and average number of key comparisons in a successful search in this table.	*CO3*	*PO1,3*	**08**
		c)	Is it possible to sort an array using Heapsort without building a heap? If yes, then Justify how?	*CO3*	*PO1,3*	**04**

******