

END SEMESTER EXAMINATION, JULY-2022

Computer Science Workshop 2 (CSE 3141)

Programme: B.Tech(CSE-A)
Full Marks: 60

Semester: 4th
Time: 3 Hours

Subject/Course Learning Outcome	*Taxonomy Level	Ques. Nos.	Marks
Analysis algorithm, using time and space complexity.	L3, L4	Q1, Q2a	8
Understanding and effectively use ADT, java collection, sorting and searching.	L1, L3	Q2b, Q2c, Q3a, b, c Q4a, b, c Q5a, b, c	22
Applying linked list, stack, queue on different problem solving.	L1, L3, L4	Q6a, b, c Q7a, b, c Q8a, b, c	18
Applying priority queue, graph on problem solving.	L1, L3, L4		
Understanding algorithm design techniques.	L1, L3, L4	Q9a, b, c Q10a, b, c	12
Applying design techniques on problem solving.	L1, L3, L4		

*Bloom's taxonomy levels: Knowledge (L1), Comprehension (L2), Application (L3), Analysis (L4), Evaluation (L5), Creation (L6)

Answer all questions. Each question carries equal mark.

1.	(a)	Write a java program to create a class ArrayWave , add a method in the class which takes an array as argument and arrange its elements in wave form such that odd elements are lesser then its neighbouring even elements.	2
	(b)	Add a method to ArrayWave that implements a recursive way to search an element using binary search.	2
	(c)	Create another class ArrayWaveApp and invoke all the methods created above. Analyse the time complexity of the created methods.	2

Note: write a single program for Q1.a, Q1.b and Q1.c

2.	(a)	Solve the following recurrence relation in terms of asymptotic notations: $T(n) = 2T(n/2) + n \log^2 n$	2
	(b)	Compare TreeSet, HashSet and LinkedHashSet classes in terms of storage, performance and order of iteration.	2
	(c)	Write a program to create a class named as HashMapDemo . Create a hash map having key as department name and value as student strength in each department. Insert four such records in the created hash map and display it.	2
3.	(a)	Write a program to create a class Policy having data members: policyId , insurerName , sumAssured , duration and premium . Define a constructor to initialize the data members with the given input. Define a displayPolicy() function to display the details of a Policy.	2
	(b)	Create another class Agent . Add a method to it which creates a LinkedList of Policy objects. Display the list. Count the number of policies present in the list	2
	(c)	Invoke the methods created above from main method in Agent class.	2

Note: write a single program for Q3.a, Q3.b and Q3.c

4.	(a)	Write a static function that takes an array of integers and returns a triplet whose sum is equal to the given value.	2
	(b)	Write a static function that takes an array in which all the elements appear even number of times except one, which appears odd number of times and returns the element which appears odd number of times in constant space and linear time complexity.	2
	(c)	Create the required class and function to execute the above static functions.	2

Note: write a single program for Q4.a, Q4.b and Q4.c

5.	(a)	Write a static function that takes an array containing 0s, 1s and 2s and sort the array so that 0s come first followed by 1s and then 2s in the end.	2
	(b)	Write a static function that takes an array of even and odd numbers and sort the array so that even numbers come first followed by odd numbers in the end in linear time.	2
	(c)	Write a static function that takes an array and sort the elements in the order of their frequency.	2
6.	(a)	Create a class Node having two data: exp (int type), coef (double type) and one link which refers to a Node type. Use a Constructor to initialize the Node and a displayNode() to display data of the node.	2

(b)	Create a class LinkedList having data members: start (a Node reference). Write a insert(int,int) function which inserts a pair of exp-coef node into the linkedlist. Define a displayList() function to display the list.	2
(c)	To the above class add a function copyListReversed() to copy the content of the linked list in another linked list in reverse order. Invoke all the methods created above from the main method.	2

Note: write a single program for Q6.a, Q6.b and Q6.c

7.	(a)	What are infix, postfix and prefix expressions. Explain with example.	2
	(b)	Create a class CStack having data members: data(char type) and top(int type) . Define push(char) and pop() function for this class. Use Constructor for initialization appropriately.	2
	(c)	Define a function doReverse() which gives the reverse of the input string by using a CStack object. Invoke this method from main method.	2
8.	(a)	Write a java program to traverse a graph using breadth first search (use ArrayDeque collection).	2
	(b)	Write a java program to implement a queue using stack. Analyse its complexity.	2
	(c)	Write a java program to reverse a queue using stack. Analyse its complexity.	2
9.	(a)	Create a BNode class to represent node of a binary search tree. Add required member variable to store string object as data part. Add other required member variables, constructor and member functions.	2
	(b)	Create another class BinarySearchTree to represent a binary search tree and add methods to insert a node and display the tree.	2
	(c)	Add a method to the BinarySearchTree class to delete a node from binary search tree having only one child node	2
10.	(a)	Create a class BinarySearchTreeApp to test the methods of BinarySearchTree .	2
	(b)	Define the various types of rotation performed to make an unbalanced tree a balanced one.	2
	(c)	Create an AVL tree using following values: 35,50,40,25,30,60,78,20,28	2

Note: write a single program for Q9.a, Q9.b, Q9.c and Q10.a

End of Questions