University of Mumbai

Examination 2020 under cluster 4 (Lead College: PCE, Panvel)

Examinations Commencing from 23rd December 2020 to 6th January 2021 and from 7th January 2021 to 20th January 2021

Program: COMPUTER ENGINEERING

Curriculum Scheme: Rev2019

Examination: SE Semester: III

Course Code: CSC303 and Course Name: DATA STRUCTURE

Time: 2 hour Max. Marks: 80

==========

Q1.	**Choose the correct option for following questions. All the Questions are compulsory and carry equal marks**
1.	Which data structure has fixed size?
Option A:	Array
Option B:	Linked List
Option C:	Graph
Option D:	Tree
2.	The result of evaluating the postfix expression 59+84-*8/
Option A:	6
Option B:	7
Option C:	5
Option D:	4

3.	What will be the output of the following program?
	
	void main ()
	{
	char str [] ="STRUCTURE";
	int len = strlen(str);
	int i;
	
	for (i=0; i<len; i++)
	push(str[i]); // pushes an element into stack
	
	for (i=0; i<len; i++
	pop (); //pops an element from the stack
	}

Option A:	ERUTCURTS
Option B:	CTURESTRU
Option C:	EUCRSTUTR
Option D:	STRUCTURE
4.	Which data structure is also known as a head tail linked list because elements can be added to or removed from the front (head) or back (tail)? However, no element can be added or deleted from the middle.

Option A:	Circular queue
Option B:	Stack
Option C:	Deque
Option D:	Priority queue
5.	A circular queue is implemented using an array of size 15. The array index starts with 0, front is 10, and rear is 14. The insertion of next element takes place at which array index?
Option A:	15
Option B:	1
Option C:	0
Option D:	11

6.	What will the output of the following function if nodes present in linked list are 6□5□2□8□9□NULL and START points the first node.
	
	void fun (struct node* START)
	{
	if (START == NULL)
	return;
	fun (START□next);
	printf ("%d “, START□data);
	}

Option A:	6,5,2,8,9
Option B:	9,8,2,5,6
Option C:	9,6,5,2,8
Option D:	9,8,2,6,5

7.	What is the output of following function if start pointing to first node of following linked list? 1□2□3□4□5□6□NULL
	
	void fun (struct node* start)
	{
	if (start == NULL)
	return;
	printf ("%d “, start□data);
	
	if (start□next! = NULL)
	fun(start□next);
	printf ("%d “, start□data);
	}

Option A:	6,5,4,3,2,1,6,5,4,3,2,1
Option B:	1,3,5,5,3,1,1,3,5,5,3,1
Option C:	1,3,5,2,4,6,1,3,5,2,4,6
Option D:	1,2,3,4,5,6,6,5,4,3,2,1
8.	Which type of linked list has no beginning and no ending.
Option A:	Circular Linked List
Option B:	Doubly Linked List

Option C:	Singly Linked List
Option D:	Multi Linked List
9.	In a doubly linked list, the number of pointers affected for an insertion operation in middle will be_____.
Option A:	1
Option B:	4
Option C:	0
Option D:	2
10.	struct node *ptr = start->next; what "ptr" will contain if it is variable of type struct node? (start points to first node)
Option A:	Address of second node
Option B:	Address field of second node
Option C:	Data of second node
Option D:	Data fields of second field
11.	What are the number of nodes in left and right sub-tree of the root node if the data is inserted in the following order in binary search tree 45, 15, 8, 56, 64, 65, 47, 12, 59, 10, 73, 50, 16, 61?
Option A:	6,7
Option B:	7,6
Option C:	8,5
Option D:	5,8

12.	Consider the following code segment in C to traverse a binary tree using the preorder
	
	void preorder (node *tree)
	{
	if (t)
	{
	Statement1
	Statement2
	Statement3
	}
	}
	The above Statements should be,

Option A:	printf("%d", tree->info);
	preorder(tree->right);
	preorder(tree->left);

Option B:	preorder(tree->left);
	preorder(tree->right);
	printf("%d", tree->info);

Option C:	preorder(tree->left);
	printf("%d", tree->info);
	preorder(tree->right);

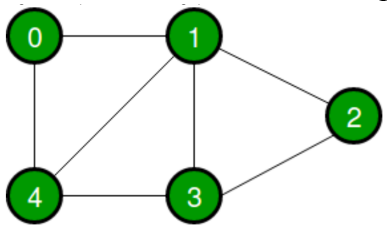
Option D:	printf ("%d", tree->info);

	preorder(tree->left); preorder(tree->right);
13.	A BST is traversed in the following order recursively: Right, root, left The output sequence will be in,
Option A:	Ascending order
Option B:	Descending order
Option C:	No specific sequence
Option D:	Random sequence
14.	What is the maximum possible number of nodes in a binary tree at level 6?
Option A:	64
Option B:	32
Option C:	48
Option D:	80

15.	Assume that a structure for a Binary Search Tree exists. What does the following function do?
	
	int function(root)
	{
	ptr = root;
	while (ptr->left!= NULL)
	{
	ptr = ptr->left;
	}
	return(ptr->data);
	}

Option A:	Leftmost child of BST
Option B:	Rightmost child of BST
Option C:	It gives error
Option D:	Root of BST
16.	When in-order and post-order traversing a tree resulted D, B, E, A, C, G, F and D, E, B, G, F, C, A respectively. the pre-order traversal would return:
Option A:	A, B, C, F, G, D, E
Option B:	A, D, E, B, C, F, G
Option C:	A, B, D, E, C, F, G
Option D:	A, B, G, F, D, E, C
17.	What is the number of edges present in a complete graph having n vertices?
Option A:	(n*(n+1))/2
Option B:	n
Option C:	(n-1)/2
Option D:	(n*(n-1))/2
18.	What is the maximum possible number of edges in a directed graph with no self-loops having 7 vertices?
Option A:	28
Option B:	35

Option C:	42
Option D:	56
19.	Using division method, in a given hash table of size 153, the key of value 172 be placed at position.
Option A:	19
Option B:	72
Option C:	17
Option D:	15
20.	What are the values of h1(k) and h2(k) in the double hashing?
Option A:	h1(k) = (m mod k) and h2(k) = 1+ (m' mod k)
Option B:	h1(k) = (1 + (m mod k)) and h2(k) = m' mod k
Option C:	h1(k) = (k mod m) and h2(k) = k mod m'
Option D:	h1(k) = (k mod m) and h2(k) = 1+ (k mod m')

Q2 **(20 Marks Each)**	**Solve any Four out of Six** **5 marks each**
A	Write a C program to test if a string is a palindrome or not using a stack data structure (Note: palindromes ignore spacing, punctuation, and capitalization)
B	Write a C program that compresses a string by deleting all space characters in the string using queue data structure
C	Give the breadth-first traversal of the graph for following graph, starting from vertex 0. Show all the steps. 
D	Consider a hash table with size = 10. Using quadratic probing, insert the keys 27, 72, 63, 42, 36, 18, 29, 101 into the table. Take c1 = 1 and c2 = 3.
E	Explain types of data structure with example
F	Write an algorithm to convert infix expression to postfix expreesion. Show stepwise execution of algorithm for converting infix expression to postfix expression for following expression A * B + C * D

Q3. **(20 Marks Each)**	**Solve any Two Questions out of Three** **10 marks each**
A	Create an AVL tree using the following data entered as a sequential set. Show all the steps. *15, 20, 24, 10, 13, 7, 30, 36, 25*. Show which rotations are used while constructing AVL tree.
B	Write a C program for Singly Linked list for performing following operations i. Create SLL ii. Display SLL iii. Delete a node from SLL

	iv. Append two SLLs
C	Draw the B-tree of order 3 created by inserting the following data arriving in sequence: 92 24 6 7 11 8 22 4 5 16 19 20 78

University of Mumbai

Examination 2020 under cluster __ (Lead College: _____)

Examinations Commencing from 23rd December 2020 to 6th January 2021 and from 7th January 2021 to 20th January 2021

Program: COMPUTER ENGINEERING

Curriculum Scheme: Rev2019

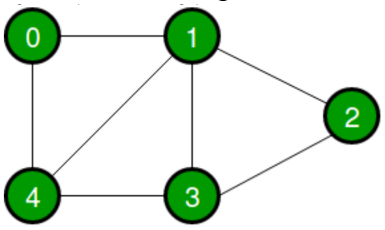
Examination: SE Semester III

Course Code: CSC303 and Course Name: DATA STRUCTURE

Time: 2 hour Max. Marks: 80

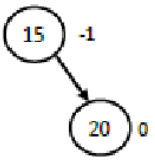
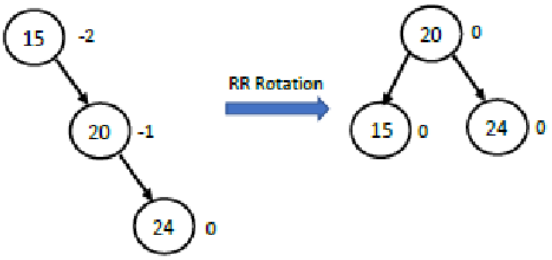
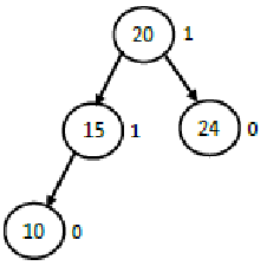
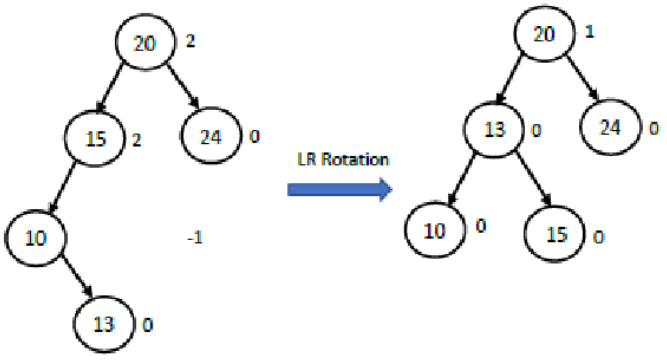
==========

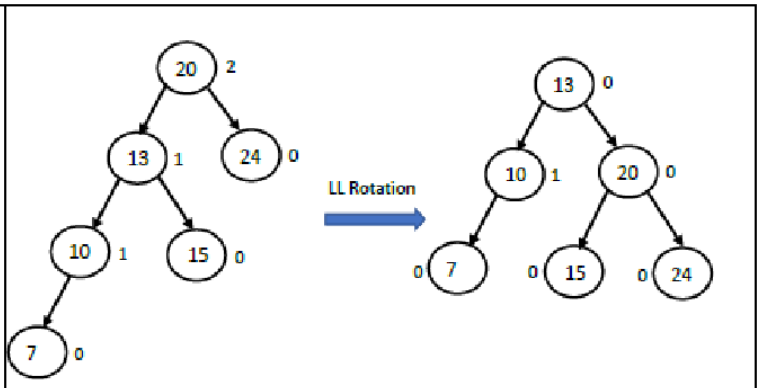
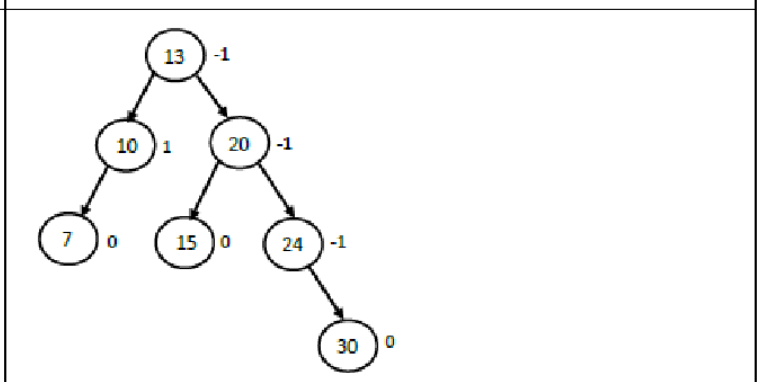
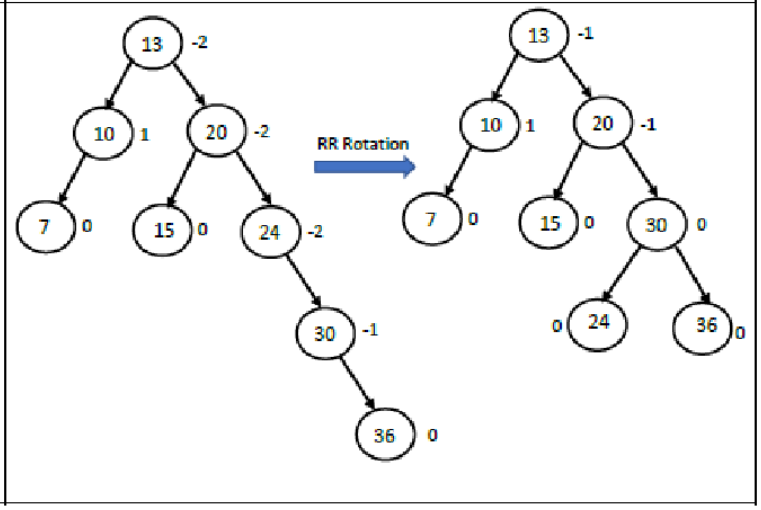
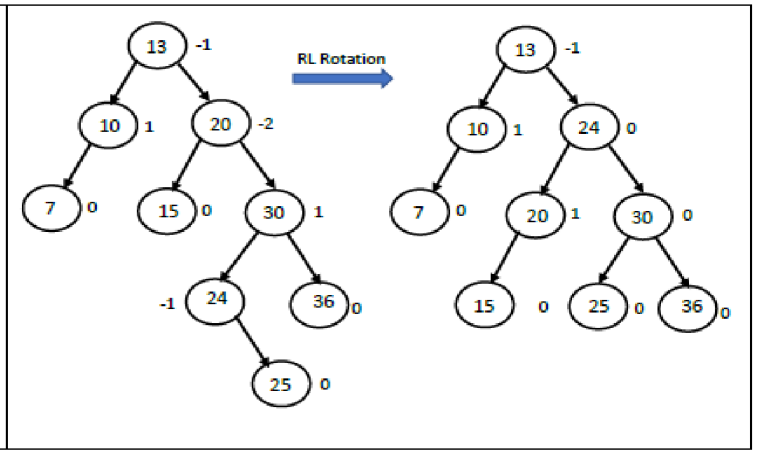
Question Number	Correct Option (Enter either 'A' or 'B' or 'C' or 'D')
Q1.	A
Q2.	B
Q3.	A
Q4	C
Q5	C
Q6	B
Q7	D
Q8.	A
Q9.	B
Q10.	A
Q11.	D
Q12.	D
Q13.	B
Q14.	A
Q15.	A
Q16.	C
Q17.	D
Q18.	C
Q19.	A
Q20.	C

Q2 (20 Marks Each)	**Solve any Four out of Six** **5 marks each**
A	Write a C program to test if a string is a palindrome or not using a stack data structure (Note: palindromes ignore spacing, punctuation, and capitalization)
	Palindrome logic/function (with main function): 3 Marks **Push operation: 1 Mark** **Pop operation: 1 Mark**
B	Write a C program that compresses a string by deleting all space characters in the string using queue data structure
	Deleting-space character logic/function (with main function): 3 Marks **Insert operation: 1 Mark** **Delete operation: 1 Mark**
C	Give the breadth-first traversal of the graph for following graph, starting from vertex 0. Show all the steps. 
	Adjacency Matrix: 1 Mark **Writing all steps with queue data structure: 3 Marks** **Final BFS traversal order: 1 Mark**
D	Consider a hash table with size = 10. Using quadratic probing, insert the keys 27, 72, 63, 42, 36, 18, 29, 101 into the table. Take c1 = 1 and c2 = 3.
	Formula of quadratic probing: 1 Mark **Writing all steps for calculating array index for given data: 3 Marks** **Final answer with number of collisions: 1 Mark**
E	Explain types of data structure with example
	Explanation of primitive and non-primitive types of data with example: 1 Mark **Explanation of linear types of data structure with example: 2 Marks** **Explanation of non-linear types of data structure with example: 2 Marks**
F	Write an algorithm to convert infix expression to postfix expreesion. Show stepwise execution of algorithm for converting infix expression to postfix expression for following expression A * B + C * D
	Algorithm for Infix to Postfix conversion: 2 Marks **Steps to convert infix to postfix expression using stack data structure: 3 Marks**

Q3. (20 Marks Each)	**Solve any Two Questions out of Three** **10 marks each**
A	Create an AVL tree using the following data entered as a sequential set. Show all the steps. *15, 20, 24, 10, 13, 7, 30, 36, 25*. Show which rotations are used.

AVL tree for the numbers *15, 20, 24, 10, 13, 7, 30, 36, 25*

Step-1: Insert 15	
Step-2: Insert 20	
Step-3: Insert 24 (Critical Node: 15)	
Step-3: Insert 10	
Step-3: Insert 13 (Critical Node: 15)	

	Step-3: Insert 7 (Critical Node: 20) 
	Step-3: Insert 30 
	Step-3: Insert 36 (Critical Node: 24) 
	Step-3: Insert 25 (Critical Node: 20) 
B	Write a C program for Singly Linked list for performing following operations

	i. Create SLL ii. Display SLL iii. Delete a node from SLL iv. Append two SLLs
	Create SLL: 2 Marks **Display SLL: 2 Marks** **Delete a node from SLL:2 Marks** **Append two SLLs: 4 Marks**
C	Draw the B-tree of order 3 created by inserting the following data arriving in sequence: 92 24 6 7 11 8 22 4 5 16 19 20 78
	Writing all steps while inserting numbers (with reason to split the node wherever required)