University of Mumbai

Examination 2020 under cluster 4 (Lead College: PCE)

Examinations Commencing from 15th June 2021 to 26th June 2021

Program: Computer Engineering

Curriculum Scheme: Rev2019

Examination: SE Semester: III

Course Code: CSC303 and Course Name: Data Structures

Time: 2 hour Max. Marks: 80

Q1.	**Choose the correct option for following questions. All the Questions are compulsory and carry equal marks**
1.	To convert the infix expression (D+(C-E)*F) into postfix, how many pop operations will be required?
Option A:	3
Option B:	4
Option C:	5
Option D:	6

2.	What is the operation performed by the following code with respect to Binary search tree, if ‘rt’ is pointing to the root node:
	`struct node *ptr=rt;`
	`struct node *fun(struct node *ptr)`
	`{`
	`        if(ptr==NULL)`
	`                return NULL;`
	`        else if(ptr->right==NULL)`
	`        return ptr;`
	`        else`
	`                return fun(ptr->right);`

Option A:	returns the smallest value in the binary search tree
Option B:	returns the right child of root node
Option C:	Returns the largest value in the binary search tree
Option D:	Returns all right nodes in the binary search tree
3.	Which of the following statements is not correct for queues?
Option A:	Queue is used in process and job scheduling
Option B:	Queue is used in depth first search traversal
Option C:	The last inserted elements is removed at the last from queue
Option D:	Elements in the queue can be removed based on their priority.
4.	The following postfix expression with single digit operands is evaluated using a stack: 2 3 ^ 4 / 7 5 + * 3 * Note that ^ is the exponentiation operator. The top two elements of the stack after ‘+’ is evaluated are:
Option A:	5,7

Option B:	7,4
Option C:	12,8
Option D:	12,2

5.	After performing these set of operations, what will be the contents of a double ended queue?
	`InsertFront(16);`
	`InsertRear(33);`
	`InsertRear(40);`
	`DeleteFront();`
	`InsertRear(25);`

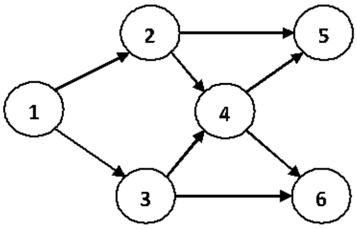
Option A:	33,40,25
Option B:	16,33,25
Option C:	16,33,40
Option D:	25,33,40
6.	Which of the following statements about stacks is incorrect?
Option A:	Stacks can be implemented using linked lists
Option B:	Stacks are first-in, first-out (FIFO) data structures
Option C:	New nodes can only be added to the top of the stack
Option D:	The last node (at the bottom) of a stack has a null (0) link

7.	What operation the following pseudo code indicates :
	`void func(Queue Q)`
	`{`
	`if(Q not empty) {`
	`int i=delete(Q);`
	`func(Q);`
	`insert(Q,i); }`
	`}`

Option A:	Reverses queue elements
Option B:	Keeps queue unchanged
Option C:	Deletes front element from queue
Option D:	Deletes all elements from queue

8.	What is the output of the following code, if linked list contains elements 16,37,28,49:
	`void fun1(struct Node* head)`
	`{`
	`if (head == NULL)`
	`return;`
	`fun1(head->next);`
	`printf(“->%d”, head->data);`
	`}`

Option A:	->16->37->28->49
Option B:	->49->28->37->16
Option C:	->37->28-.>49->16
Option D:	->28->49->37->16

9.	How many pointers are contained as data members in the nodes of a circular, doubly linked list of integers with seven nodes?
Option A:	7
Option B:	8
Option C:	14
Option D:	15
10.	Which is not the property of Linear data structures ?
Option A:	Contiguous allocation
Option B:	Sequential access
Option C:	Static or dynamic allocation
Option D:	Abstract Data type
11.	Consider the DAG with Consider V = {1, 2, 3, 4, 5, 6}, shown below. Which of the following is not a breadth first search sequence for the graph? 
Option A:	1 2 3 4 5 6
Option B:	1 3 2 4 6 5
Option C:	1 3 2 6 4 5
Option D:	3 2 4 1 6 5
12.	A binary search tree is created by inserting the numbers 2, 6, 0, 1, 9, 8, 4, 7, 3, 5. What is the post-order traversal sequence of the resultant tree?
Option A:	0 1 2 3 4 5 6 7 8 9
Option B:	0 2 4 3 1 6 5 9 8 7
Option C:	1 0 3 5 4 7 8 9 6 2
Option D:	1 0 3 4 5 6 7 8 9 2

13.	What the following code do:
	ptr=head;
	while(ptr!=NULL) {
	  tr=ptr->next->next; }

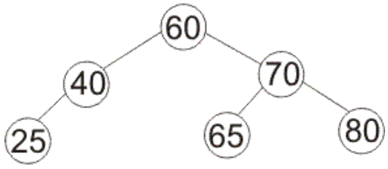
Option A:	Traverse list
Option B:	Traverse even position nodes
Option C:	Traverse odd position nodes
Option D:	Deletes odd position nodes

14.	Select the operation performed by the following code segment with respect to binary tree:
	
	`void  func(struct  Node* p)`
	`{`

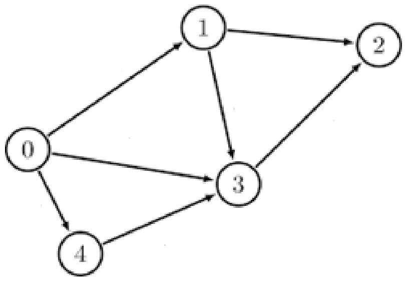
```
        if  (p == NULL)
            return;
        else
        {
            struct  Node* temp;

            func(p->left);
            func(p->right);

            temp     = p->left;
            p->left = p->right;
            p->right = temp;
        }
    }
```

Option A:	find the minimum element in a binary search tree
Option B:	find the maximum element in a binary search tree
Option C:	Interchange of nodes
Option D:	Converts tree into its mirror image
15.	If you insert 75 into the following binary search tree using the algorithm that keeps the tree height-balanced by doing rotations, what tree do you get? 
Option A:	Left child of 65
Option B:	Right child of 65
Option C:	Right child of 40
Option D:	Left child of 80
16.	How many nodes will be created in a B-tree by inserting the keys : 11,14,17,20,27,31,41,29,75,30 (Assume ORDER 5) ?
Option A:	4
Option B:	5
Option C:	6
Option D:	7
17.	Which of the following statement is incorrect with respect to graphs?
Option A:	A sequence of vertices that connect two nodes in a graph is called a path.
Option B:	Degree of vertex in a graph is the number of edges that touch it.
Option C:	A tree is a graph with cycles.
Option D:	In complete graph, every vertex is directly connected to every other vertex
18.	What is the worst case for linear search?
Option A:	Search key is available at first location
Option B:	Search key is available at last location
Option C:	Search key is available at middle of array
Option D:	Search key is available anywhere in the array

19.	In a Doubly linked list with 2 pointers namely, 'prev' and 'next', and a pointer 'Temp' pointing to some node except first or last node, which of the following statement will delete the element pointed by 'Temp'?
Option A:	Temp->prev->next=Temp->next ; Temp->next->prev=Temp->prev; free(temp);
Option B:	Temp->prev->next=Temp->prev ; Temp->next->prev=Temp->next; free(temp);
Option C:	Temp->prev->prev=Temp->next ; Temp->next->next=Temp->prev; free(temp);
Option D:	Temp->prev->prev=Temp->prev ; Temp->next->next=Temp->next; free(temp);
20.	Max .no. of nodes in a binary tree with level 6 are
Option A:	32
Option B:	63
Option C:	64
Option D:	31

Q2	**Solve any Four out of Six 5 marks each**
A	Consider marks of 5 subjects of a student represented as singly linked list. Write a C program to compute the total and percentage of the student.
B	An array contains the elements – 8,13,17,26,44,56,88,97. Using binary search algorithm, trace the steps followed to find numbers 56 & 9. At each step, show the contents of low, high & mid and array after each iteration
C	Create a Binary Search Tree for the following sequence and write all the 3 traversal sequences from resultant BST: 45,39,56,12,34,78,32,10,89,54,67,81.
D	Use linear probing, insert the following keys in a hash table of size 11: 15,85,90,54,67,43,76. Find the number of collisions.
E	Illustrate topological sorting for the following graph: 
F	Define circular queue. Assume a circular queue with a capacity 6, currently having the elements 50 and 70 at locations 2 and 3 respectively. Show with example, the queue full and queue empty conditions by performing necessary operations on circular queue.

<table>
<tr><th>Q3.</th><th>Solve any Two Questions out of Three 10 marks each</th></tr>
<tr><td>A</td><td>Create a AVL tree for the sequence:
I, N, F, O, R, M, A, T, G.
Consider the characters to arrange in alphabetic sequence.
Show the tree after each insertion with balance factors.</td></tr>
<tr><td>B</td><td>Given the following frequencies for characters, find the Huffman code for all the characters:
<table><tr><td>Character</td><td>S</td><td>T</td><td>I</td><td>N</td><td>G</td></tr></table>
</td></tr>
</table>

<table>
<tr><td></td><td><table><tr><td>Frequency</td><td>9</td><td>16</td><td>2</td><td>30</td><td>12</td></tr></table></td></tr>
<tr><td>C</td><td>Define recursion. Differentiate between iteration and recursion. Write a C program to check whether a string is palindrome or not, with the help of stack data structure.</td></tr>
</table>

University of Mumbai

Examination 2020 under cluster 4 (Lead College: PCE)

Examinations Commencing from 15th June 2021 to 26th June 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.	C
Q2.	C
Q3.	B
Q4	D
Q5	A
Q6	B
Q7	B
Q8.	B
Q9.	C
Q10.	A
Q11.	D
Q12.	C
Q13.	B
Q14.	D
Q15.	D
Q16.	A
Q17.	C
Q18.	B
Q19.	A
Q20.	B

Q2 (20 Marks Each)	**Solve any Four out of Six** **5 marks each**
A	Consider marks of 5 subjects of a student represented as singly linked list. Write a C program to compute the total and percentage of the student.
	Node / Variables declaration: 1 Mark **Program logic and functions: 4 Marks**
B	An array contains the elements – 8,13,17,26,44,56,88,97. Using binary search algorithm, trace the steps followed to find numbers 56 & 9 . At each step, show the contents of low, high & mid and array after each iteration

i) Item to be searched=56

	low			mid				high
Initial Array	**8**	**13**	**17**	**26**	**44**	**56**	**88**	**97**
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**

Low=0; high=7; mid=3; a[mid]=a[3]=26
26 < 56,

					low	mid		high
Step1	8	13	17	26	**44**	**56**	**88**	**97**
	0	1	2	3	**4**	**5**	**6**	**7**

Low=mid+1=4; high=7; mid=5; a[mid]=a[5]=56
Item is found, so return location 5.

ii) Item to be searched=9

	low			mid				high
Initial Array	**8**	**13**	**17**	**26**	**44**	**56**	**88**	**97**
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**

Low=0; high=7; mid=3; a[mid]=a[3]=26
26 > 9,

	low	mid	high					
Step1	**8**	**13**	**17**	26	44	56	88	97
	0	**1**	**2**	3	4	5	6	7

low=0; high=mid-1=2; mid=1; a[mid]=a[1]=13
13 > 9,

	low, mid, high							
Step2	**8**	**13**	**17**	26	44	56	88	97

<table>
<tr><td></td><td>

______ **0** 1 **2** 3 4 5 6 7

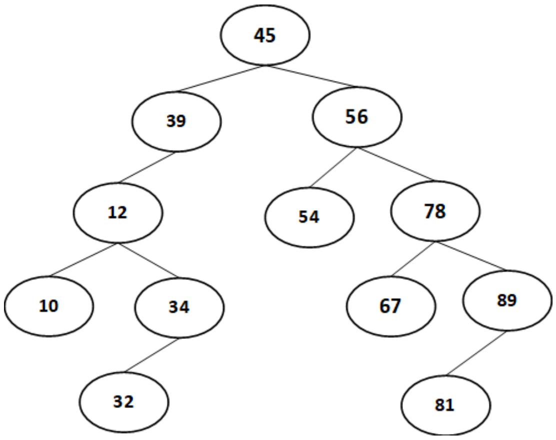
low=0; high=mid-1=0; mid=0; a[mid]=a[0]=8
8< 9,
Low=mid+1=1; high=0;
Since (high < low) , this means the item is not found in the array.

</td></tr>
<tr><td>C</td><td>

Create a Binary Search Tree for the following sequence and write all the 3 traversal sequences from resultant BST: 45,39,56,12,34,78,32,10,89,54,67,81.

</td></tr>
<tr><td></td><td>

Creation of BST: 2 Marks
Writing all 3 traversals correctly: 3 Marks



Inorder Traversal: 10,12,32,34,39,45,54,56,67,78,81,89
Preorder Traversal: 45,39,12,10,34,32,56,54,78,67,89,81
Postorder Traversal: 10,32,34,12,39,54,67,81,89,78,56,45

</td></tr>
<tr><td>D</td><td>

Use linear probing, insert the following keys in a hash table of size 11:
15,85,90,54,67,43,76.
Find the number of collisions.

</td></tr>
<tr><td></td><td>

Formula of linear probing: 1 Mark
Writing all steps for calculating array index for given data: 3 Marks
Final answer with number of collisions: 1 Mark

Final Hash Table is:

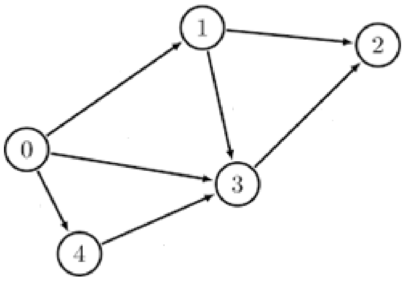
<table>
<tr><td>43</td><td>67</td><td>90</td><td>76</td><td>15</td><td></td><td></td><td></td><td>85</td><td></td><td>54</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr>
</table>

Number of collisions=5

</td></tr>
<tr><td>E</td><td>

Illustrate topological sorting for the following graph:

</td></tr>
</table>



Writing in-degrees in all steps correctly and removing nodes and adding in the topological sorting order is expected.

	Vertex 0	Vertex 1	Vertex 2	Vertex 3	Vertex 4	Output Sequence	Action
In-degree	0	1	2	3	1	0	Delete Vertex 0 & update in-degrees
	-	0	2	2	0	0,1	Delete Vertex 1 & update in-degrees
	-	-	1	1	0	0,1,4	Delete Vertex 4 & update in-degrees
	-	-	1	0	-	0,1,4,3	Delete Vertex 3 & update in-degrees
	-	-	0	-	-	0,1,4,3,2	Delete 2

Topological order is: 0,1,4,3,2

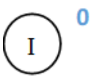
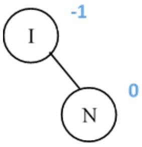
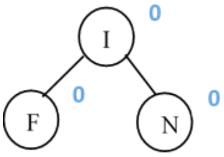
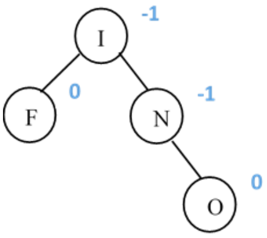
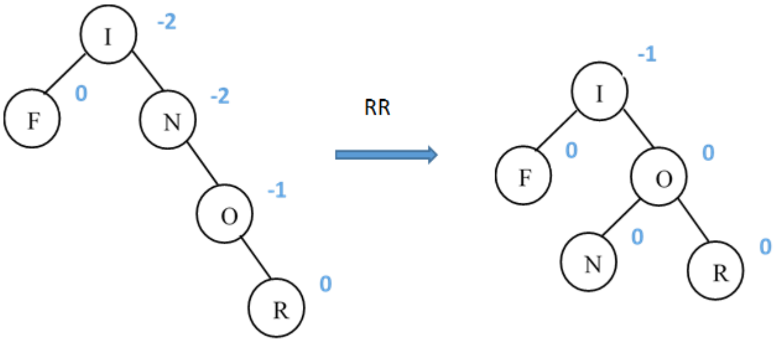
OR
In case if 4 is considered before 1 in sequence, then the oder changes and Topological order is: 0,4,1,3,2

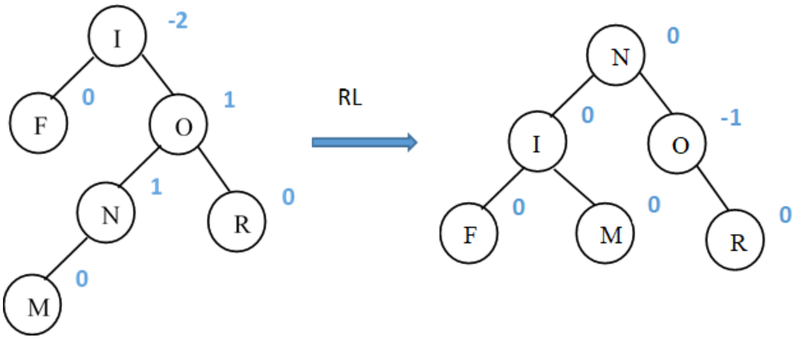
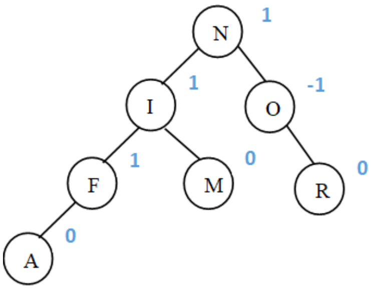
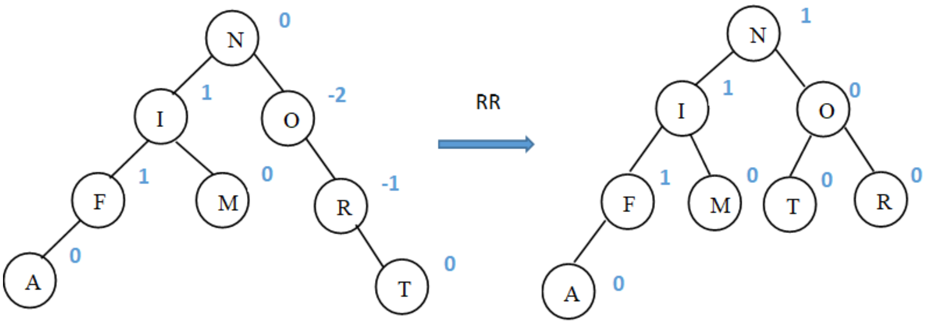
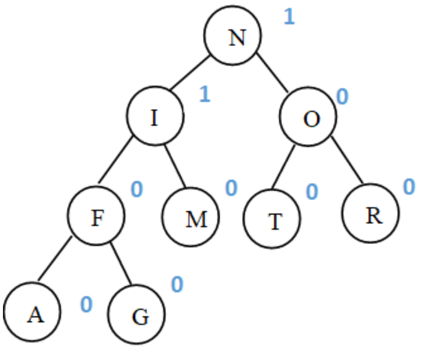
F

Define circular queue. Assume a circular queue with a capacity 6, currently having the elements 50 and 70 at locations 2 and 3 respectively. Show with example, the queue full and queue empty conditions by performing necessary operations on circular queue.

Definition of circular queue: 2 Marks
Initial queue status with front and rear pointers pictorial representation: 1 Mark
Queue full illustration with insertion operations: 1 Mark
Queue empty illustration with deletion operations: 1 Mark

Q3. **(20 Marks Each)**	**Solve any Two Questions out of Three** **10 marks each**
A	Create a AVL tree for the sequence: I, N, F, O, R, M, A, T, G. Consider the characters to arrange in alphabetic sequence. Show the tree after each insertion with balance factors.

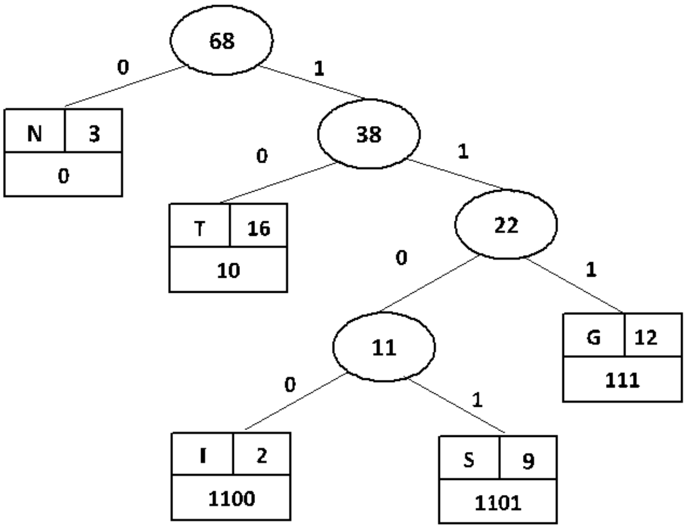
Insert I:	
Insert N:	
Insert F:	
Insert O:	
Insert R:	 I -2 F 0 N -2 O -1 R 0 RR I -1 F 0 O 0 N 0 R 0

Insert M:	
Insert A:	
Insert T:	
Insert G:	

B

Given the following frequencies for characters, find the Huffman code for all the characters:

Character	S	T	I	N	G
Frequency	9	16	2	30	12

	Selection of proper nodes and combining: 6 Marks **Writing binary codes: 2 Marks** **Correct computation of Huffman codes: 2 Marks**  68 0 1 N 3 0 38 0 1 T 16 10 22 0 1 G 12 111 11 0 1 I 2 1100 S 9 1101
C	Define recursion. Differentiate between iteration and recursion. Write a C program to check whether a string is palindrome or not, with the help of stack data structure.
	Definition of recursion: 2 Marks **Difference between iteration and recursion: 2 Marks** **Program logic and main function: 2 Marks** **pop operation: 2 Marks** **push operation: 2 Marks**