|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **U.S.N.** |  |  |  |  |  |  |  |  |  |  |

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

**Autonomous Institute Affiliated to VTU**

**July / August 2019 Supplementary Examinations**

|  |  |
| --- | --- |
| **Programme: B.E.** | **Semester : III** |
| **Branch : Computer Science and Engineering** | **Duration: 3 hrs.** |
| **Course Code: 15CS3DCDST** | **Max Marks: 100** |
| **Course: Data structure** | **Date: 27.07.2019** |

**Instructions**: 1. Answer any FIVE full questions, choosing one full question from each unit.

2. Missing data, if any may 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** |  |
| 1 | a) | i. Show the tracing for tower of Hanoi problem for n=3 with an algorithm  ii. Explain with an example about dynamic memory allocation methods. | **08** |
|  | b) | Write an algorithm/program for postfix expression and Demonstrate the evaluation of following given postfix expression.  2 3 8 \* + 4 48 4 2 + / 6 \* + - - | **08** |
|  | c) | Analysis the code and print the output?  #include<stdio.h>  int x;  void Q(int z)  {  z += x;  printf("%d",z);  }  void P(int \*y)  {  int x = \*y+2;  Q(x);  \*y = x-1;  printf("%d",x);  }  void main(void)  {  x=5;  P(&x);  printf("%d",x);  } | **04** |
|  |  | **UNIT - II** |  |
| 2 | a) | Write a ‘C’ program to implement Stack primitive operations using singly link list | **10** |
|  | b) | Write a ‘C’ program to implement doubly link list and perform the following operations:   1. Insert a new node to the end of the double linked list. 2. Delete the node of a given data. | **10** |
|  |  | **OR** |  |
| 3 | a) | Write a ‘C’ program to implement Single Link List and perform the following operations   1. Create a new node at the end of linked list. 2. Insertion of a node at any position 3. Deletion of last node in the list. | **10** |
|  | b) | Write a ‘C’ program to implement Single Link List and perform the following operations   1. Sort the linked list. 2. Reverse the linked list. 3. Concatenation of two linked lists | **10** |
|  |  | **UNIT - III** |  |
| 4 | a) | Construct a Binary Tree with the given data:   1. Preorder: 8,5,9,7,1,12,2,4,11,3,   Inorder: 9, 5, 1, 7, 2, 12, 8, 4, 3, 11  Note: Traversal for Binary Tree   1. Postorder: 10,9,23,22,27,25,15,50,,95,60,40,29   Inorder: 9, 10, 15, 22, 23, 25, 27, 29, 40, 50, 60, 95  Note: Traversal for Binary Search Tree | **10** |
|  | b) | Write a ‘C’ program to implement the following operation in Binary Search Tree   1. Insert a node 2. inorder 3. preorder 4. postorder | **10** |
|  |  | **UNIT - IV** |  |
| 5 | a) | Illustrate splitting, merging and redistribution of keys in B-Tree (Consider any B-tree order) with an example. | **08** |
|  | b) | Consider an AVL tree given below fig.1 and insert the following data: 18, 81, 29, 15, 19, 25, 26, 1.  Also delete nodes 39, 63 from the AVL tree formed after inserting the above data.  C:\Documents and Settings\Syed Akram\Local Settings\Temporary Internet Files\Content.Word\WhatsApp Image 2018-11-28 at 6.45.50 PM.JPEG  Fig1: AVL Tree | **12** |
|  |  | **OR** |  |
| 6 | a) | Discuss the use of Threaded Binary Tree (TBT). Explain the different types of TBT with an example. | **08** |
|  | b) | Consider a sequence of numbers from 1 to 7 as the values of nodes in a splay tree, show the steps involved in splaying the tree at node 1. | **05** |
|  | c) | Explain Red-Black tree insertion with an example data 47, 32, 71, 93, 65, 82, 87 | **07** |
|  |  | **UNIT - V** |  |
| 7 | a) | Write the insertion sort algorithm. | **05** |
|  | b) | Given a hash table of size 13, show the contents of your hash table after inserting the values {8, 2, 7, 18, 15, 19, 23, 15, 20, 16}, show any collision happens during insertion and explain how it resolved using linear and quadratic probing.  Note : [ Hash function is K % M ], K is Key and M is Table size. | **10** |
|  |  | c) | Demonstrate sorting by counting for the values 4, 6,1,4,1,7,8,2 assuming the values are in the rays 0 to 9. | **05** |

**\*\*\*\*\*\*\***