11 MAY 2018

Sheet No. 1

B.Tech. 02/2018

NATIONAL INSTITUTE OF TECHNOLOGY, KURUKSHETRA
THEORY EXAMINATION

Question Paper

Month and Year of the Examination: **May/June-2018**

Programme: **B.Tech.** Semester: **2nd**

Subject: **Data Structures**

Course No: **CSPC-12** Maximum Marks: **50**

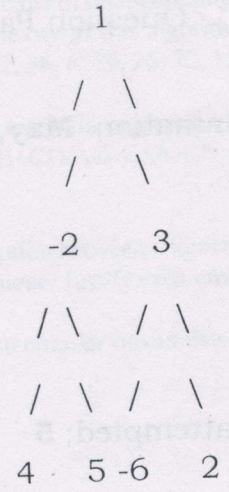
Number of Questions to be attempted: **5** Time allowed: **3 Hours**

Total No. of Questions: **5** Total No. of Pages used: **2**

The candidates, before starting to write the solution, should please check the question paper for any discrepancy, and also ensure that they have been delivered the question paper of right **course no**. and **subject title**. Assume suitably and state, additional data required, if any.

Note: All Questions are Compulsory

1.	(a)Q: Discuss the classifications of various data structures proposed in C language.	**3**
	(b)Q: Write a clear difference with an example between **malloc** and **calloc** function used in C language.	**2**
	(c)Q: WAP in C language to sort n elements using Quick Sort.	**5**
2.	(a)Q: WAP in C language to count the number of characters in a given file.	**5**
	(b)Q: WAP in C language to implement two stacks in a single array.	**5**
3.	(a)Q: What are the advantages and drawbacks of singly linked list?	**2**
	(b)Q: WAP in C language to check that a given matrix is sparse matrix or not.	**3**
	(c)Q: WAP in C language to convert a given infix expression to postfix expression.	**5**
4.	(a)Q: WAP in C language to reverse every group of k nodes in a given double linked list.	**5**
	(b)Q: When does using a doubly linked list seem to be best option in real life scenario? Can you suggest the practical uses of it?	**2.5**
	(c)Q: What are some real-world applications of a queue data structure?	**2.5**

5.	(a)Q: WAP in C language to find the largest subtree sum of a given tree below (Note: create the tree first using create function).  (b)Q: WAP in C language to implement a heap sort for decreasing order using min heap for following input and output values: Input: arr[] = {5, 3, 10, 1} Output: arr[] = {10, 5, 3, 1}	**5**

**********End **********