Roll Number:________________________

Thapar University, Patiala

Department of Computer Science and Engineering

MST B. E. COE (Second Year): Semester-IV	Course Code: UCS406 Course Name: Data Structures and Algorithms
March 18, 2016	Friday, 10.30 – 12.30 Hrs
Time: 2 Hours, M. Marks: 20	Name of Faculty: DG, RIR, TBH, ANK

Note: ***All questions are compulsory and attempt all parts of a question at one place.***
Assume missing data, if any, suitably. Clearly specify your notations used in algorithms.

Q.1 (a)	**Write an algorithm** to sort numbers using quick-sort and **illustrate the working** of algorithm for following set of numbers: 5, 2, 25, 17, 8, 14	(3)
Q.1 (b)	**Convert** following infix expression into postfix expression using stack as an intermediate data structure. Also, **evaluate** the resulting postfix expression using stack. Infix Expression: 5 * 4 + (30 – 50/2)	(2)
Q.2 (a)	Given a list of elements with priorities: (a, 7), (b, 13), (c, 10), (d, 11), (e, 21), (f, 17) (i) **Build** a max-heap on the basis of priority (ii) **Remove** the element with highest priority from the heap, showing all intermediate steps. (iii) **Insert** a new element ‘g’ with priority ‘15’ in the heap and show resultant max-heap.	(3)
Q.2 (b)	Given an array F with size n. Assume the array content F[i] indicates the length of the i^{th} file and we want to merge all these files into one single file. Assume individual files are sorted. **Write a greedy algorithm** that gives optimal solution to merge these files. **Illustrate the working** of algorithm for following example: F [10, 15, 100, 50, 20]	(2)
Q.3 (a)	Given a singly linked list, **write a function to reverse every k nodes** (where k is an input to the function) and returns the modified linked list. If the number of nodes in a list is not a multiple of k then left-out nodes in the end should remain as it is. You have to retain the memory address of the nodes without modifying it i.e. you can’t just interchange the values in the nodes. Only constant memory is allowed. For example: Inputs : 1->2->3->4->5->6->7->8-> null and k = 3 Output: 3->2->1->6->5->4->7->8-> null	(3)
Q.3 (b)	N people have decided to elect a leader by arranging themselves in a circle and eliminating every M^{th} person around the circle, closing ranks as each person drops out. **Write an algorithm** using circular linked list to find which person will be the last remaining and elected as a leader.	(2)
Q.4 (a)	(i) **Develop a BST** by inserting nodes from the following sequence one by one. 62, 32, 22, 57, 37, 47, 42, 72, 92, 82 (ii) **Write an algorithm** for deleting a node with particular value in a BST. You need to also write complete pseudocode of any subroutine used. Also, show the resultant tree after deleting node with value 32.	(3)

Q. 4 (b)	(i) **Calculate and justify** the complexity of following function.
	void fun(int n, int arr[])
	{
	int i = 0, j = 0;
	for(; i < n; ++i)
	{
	while(j < n && arr[i] < arr[j])
	j=j*2;
	}
	}
	
	(ii) State whether following statements are **True/False**. **Justify** your answer.
	I. If I prove that an algorithm takes $\Theta(n^2)$worst-case time, it is possible that it takes O(n) on all possible inputs?
	II. If I prove that an algorithm takes $O(n^2)$ worst-case time, it is possible that it takes O(n) on some inputs?
		(2)