BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI HYDERABAD CAMPUS,
Data Structures and Algorithms
CS F211 / IS F211

Homework Assignment – 6

1. Write a program to generate 1000, 100000, 100000, 200000, 500000, and 1000000 numbers randomly (use srand() and rand()) and sort them using the following algorithms. Assume the integers are bound in the range [0, 10000000] for all these questions. Measure the time taken to run each of the algorithm and print it (use time() and clock() functions). You are free to define the number and size of the bucket in case of bucket sort.
   a. Count sort
   b. bucket sort
   c. radix sort

2. A majority element in an array A[] of size n is an element that appears more than n/2 times (and hence there is at most one such element). Write a function which takes an array and emits the majority element (if it exists), otherwise prints NONE in O(n) as follows:
   Input1:
   9
   3 3 4 2 4 4 2 4 4
   Output1: 4

   Input2:
   9
   3 3 3 4 2 4 4 2 4
   Output2: NONE

3. Let A be an array of n distinct positive integers. If i < j and A[i] > A[j] then the pair (i, j) is called an inversion of A. Given n and an array A your task is to find the number of inversions of A in O(nlogn). (**Note :** You cannot use global variables.)

4. A contractor is given a job to renovate the old building. Each day's work has been decided and the according to work, each day **distinct** number of workers are hired for that day. The contractor is interested in knowing that for each day, how many days in future are there when number of workers required is less than the number of workers on the current day. You should solve this problem in O(nlogn)
   Input Format:
   N

N **distinct** numbers separated by space each number representing the no. of workers required for that day.

Constraints: N ≤ 1000 and noOfWorkers[i] ≤ 100

output: a line contains N integer separated by a space, where i^{th} integer represents the number of days of future, when no. of workers is less than the current day.

Input :

7

5 8 1 6 3 2 7

output : 3 5 0 2 1 0 0

Explanation : For day-1, 5 workers are there and in future only on day-3, day-5, day-6 no of workers are 1, 3, 2 < 5 hence 3 in output for day-1 and similarly for other days.

5. Given two arrays A1[] and A2[], sort A1 in such a way that the relative order among the elements will be same as those are in A2. For the elements not present in A2, append them at last in sorted order. Your program should handle all cases like number of elements in A2[] may be more or less compared to A1[]. A2[] may have some elements which may not be there in A1[] and vice versa is also possible.

Example:

Input: A1[] = {2, 1, 2, 5, 7, 1, 9, 3, 6, 8, 8}

    A2[] = {2, 1, 8, 3}

Output: A1[] = {2, 2, 1, 1, 8, 8, 3, 5, 6, 7, 9}

6. Write a menu driven program to create a Singly Linked List by giving option a, b, c for following operation by appropriately reading a number X to be inserted into the list:
   a. Insert at beginning of the List (write a function InsertAtBeg() and call from main)
   b. Insert at end of the List(write a function InsertAtEnd() and call from main)
   c. Read another number Y before which X is to be added.(write a function Insert() and call from main)
   (**Note:** You can't define global variables except your structure.)

7. Write a menu driven program to create a Singly Linked List by inserting in the beginning only i.e. option a and giving option b, c, d for following operation:
   a. Insert at beginning of the List
   b. Delete at beginning of the List.
   c. Deleting from the end of the List.
   d. Read a number X which is to be added.
   (**Note**: In case list is empty proper error message (Underflow) should be displayed. You can't define global variables except your structure.)

8. Write a **mergeSortedList**() function that takes two lists, each of which is sorted in increasing order, and merges the two together into one list which is in increasing order. **mergeSortedList** () should return the new list. You can't define global variables except your structure.

   For Example : if the first linked list a is 5->10->15->NULL and the other linked list b is 2->3->20->NULL, then **mergeSortedList** () should return a pointer to the head node of the merged list 2->3->5->10->15->20->NULL
   (**Note:** You need to print the modified list and for creating the two list make function InsertAtBeg to insert at beginning which will very easily create a list)

9. Create a Linked List and reverse the Linked-List(original linked-list should be reversed not a new one should be created) and print the reversed-list. So for creating Linked-List simply insert at beginning of linked-list and then reverse the Linked-List created and after reversing print the Linked-List. Use proper functions like InsertAtBeg(), reverse() and printList() to solve this problem. You can't define global variables except your structure.

   Ex : your List is   4 -> 5 ->1 ->3 -> NULL

   then your reversed list is 3->1->5->4->NULL

10. Create a Linked List and reverse the Linked-List(original linked-list should be reversed not a new one should be created) through **Recursion** and print the reversed-list. So for creating Linked-List simply insert at beginning of linked-list and then reverse the Linked-List created and after reversing print the Linked-List. Use proper functions like InsertAtBeg(), reverse() and printList() to solve this problem. You can't define global variables except your structure.

    Ex : your List is   4 -> 5 ->1 ->3 -> NULL

    then your reversed list is 3->1->5->4->NULL

--------------------------------XXXXXXXXXXXXXXXXXXXXXXXXX--------------------------------