INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667
Data Structures (CS 102), B.Tech CSE and ECE

Assignment: 4 Spring Semester 2015-16

1. Write a C++/JAVA program to sort a linked list.

2. Implement a version of bubble sort that builds up the sorted array from largest to smallest.

3. Implement radix sort algorithm using arrays. Deduce the time complexity *T(n)* for the best, worst and average cases.

4. Implement Quick Sort in separate recursive (QR) and iterative (QI) versions.

(ii) Implement pivot selection techniques and other performance enhancements as applicable for each.

(iii) Compare performance of QR and QI on a significant sized collection of randomly generated inputs. Choose based on this comparison the better of QR and QI say QS.

(iv) Implement Selection Sort iteratively (SS).

(v) Plot performance of QS versus SS for various sizes (lengths) of input data (lists). Find the cutoff size, say N, below which SS outperforms QS.
Repeat the measurements at different system loads to get remove spurious variations.

(vi) Rewrite QS such that when size of sublist becomes less than N, SS is invoked. Call this version QI.

(vii) Add the performance plot of QI to the one obtained in (v) and measure the performance improvement (as compared to QS).

(viii) Find a system implementation of Quick Sort and add this to the plot obtained in QI.

5. Transform each of the following postfix expressions to infix (using operand stacks)

(i) AB+C-
(ii) ABC+-
(iii) AB-C+DEF-+$
(iv) ABCDE-+$*EF*-
(v) AB+C*DE--FG+$

6. Convert the following infix operation to prefix using **<u>operator stack</u>**.

$$x * y /(z + a)\$b + c * d /(e + f)\$(g - h) / k - l$$

7. The Tower of Hanoi problem involves a stack of *n* graduated disks and a set of three needles called A, B and C. The initial setup places the *n* disks on needle A. The task for the player is to move the disks one at a time from needle to needle until the process rebuilds the original stack, but on needle C. The challenge is the fact that at no time a larger disk can be placed on top of a smaller disk.

Your task is to develop a program that demonstrates the Tower of Hanoi problem. A simple demo for this problem can be found on the following web page

http://mathworld.wolfram.com/TowerofHanoi.html

Use the algorithm that you have seen in the lecture class for the implementation of this problem.

8. Find the recurrence relation for Quick sort algorithm for both Worst and Best cases. Deduce the recurrence relation and find the time complexity in big-O notation.

9. Find the output of the following code: Show the content of the stacks before printing the output.

```
#include <iostream.h>
 int hard(int n)
    {
         if (n< 0) return;
         cout<<n-2;
         hard(n-2);
         cout<<n-1;
         hard(n-1);
         cout<<n;
         hard(n-3);
   }

   void main()
   {cout<<hard(6);}
```

10. Write a C++/Java function to delete the bottom element of the stack.