**Data Structure Problem Set A**

**Linked List**

1. Given a pointer to the head of a singly linked list, write a function to insert a new node at the very end of the linked list. The head of the linked list could also be NULL i.e. the linked list could initially be empty.
2. Given pointer to the head of a singly linked list, write a function to add a node at the beginning of the list.
3. Given a pointer to the head of a singly linked list and a number n, delete the node at nth position in the linked list.
4. Given a singly linked list of integers, print the elements of linked list in reverse order.
5. Given pointer to the head of a linked list and a number n = desired position, insert an element at position n from the beginning in the linked list.
6. Given a Singly linked list as input, reverse it.
7. Given an unsorted linked list of integers, sort it in non-decreasing order and display the linked list.
8. Compare two linked list of integers to verify whether they are identical. The linked lists are identical if they have the same size and same elements in same order.
9. Given two sorted linked lists, merge these two lists to create one single sorted linked list.
10. Given a singly linked list of integers and a number n, write a program to return nth node from the end of the linked list.
11. Given a singly linked list of integers that is sorted, remove all duplicates from the list.
12. Given a linked-list as input, verify whether it contains a loop.
13. Two singly linked lists of integers with different number of elements and different start nodes merge after some nodes (possibly after different number of nodes from their respective heads) at a Y-junction. Return the data in the node at which they merge. Just print the value of the first common node or the merge point.
14. Given a Singly linked list, rotate the list to right/left by K places where K is a non-negative integer.
15. Given a singly linked list of integers with N nodes, if we mark nodes in the list at positions 1 to N, remove nodes at positions that are multiple of a given number K.
16. Create a doubly link list of integers by inserting the nodes in sorted order.
17. Given a double linked list of integers as input, reverse the list.

**Stack**

1. Given an expression only containing upper-case alphabets: A-Z, operators: /, \*, +, - and characters: (, ), {, }, [, ] are used to mark start and end of parentheses, validate whether the parentheses are balanced or not.
2. As the problem title says, you are given a string S in infix form which you need to convert to postfix form. The expression contains operands from the set of uppercase English alphabets, operators from the set {+, -, \*, /}. Also there may be rounded parentheses '()' in the expression. There will be no other character in the string. The operators have the following properties:
   * Expressions within brackets are always evaluated first.
   * Multiplication (\*) and Division (/) have the same precedence. Similarly, Addition (+) and Subtraction (-) have same precedence. But, + and / have higher precedence than + and -.
   * All operators have left to right associativity.
3. Evaluate postfix expressions:
   * Input:  
     2  
     24+  
     342+/
   * Output:  
     6.000000  
     0.500000
4. Albert is a school teacher and he is correcting the examination answer sheets of his class. He is getting bored and so decides to play a game. He has three options in this game:
   * He can choose to correct a new answer sheet. After correcting it and giving it a score, he places this answer sheet on the top of the pile.
   * He can remove the answer sheet that is on the top of the pile provided there is at least one answer sheet in the pile.
   * He may want to know who has scored the minimum of all students whose answer sheets are in the pile.
   * Input:  
     1  
     7  
     1 3  
     1 -1  
     1 7  
     3  
     2  
     2  
     3
   * Output:  
     -1  
     3