

CSE221 Programming Assignment 2

Due: 10/6 at 4:00pm

Instruction: zip your source code as assignment1_yourStudentID.zip and submit it via Blackboard. Please make sure your code is runnable, otherwise you will get no point since we cannot debug your code for grading. We will use C++ 11 and your code should be compiled in Linux. Please do not use any STL libraries unless it is mentioned. Also please provide how to use your code in main() as examples.

1. (30 points) Implement the stack ADT in a fully generic manner (through the use of templates) by means of a singly linked list. (Give your implementation “from scratch,” without the use of any classes from the Standard Template Library or any other data structures presented in this course.)

2. (35 points) Write a simple text editor, which stores a string of characters using the list ADT, together with a cursor object that highlights the position of some character in the string (or possibly the position before the first character). Your editor should support the following operations and redisplay the current text (that is, the list) after performing any one of them.

- left: Move cursor left one character (or nothing if at the beginning)
- right: Move cursor right one character (or do nothing if at the end)
- delete: Delete the character to the right of the cursor (or do nothing if at the end)
- insert c : Insert the character c just after the cursor

3. (35 points) Write a program that takes, as input, a fully parenthesized, arithmetic expression and converts it to a binary expression tree. Your program should display the tree in some way and also print the value associated with the root. For an additional challenge, allow for the leaves to store variables of the form x_1, x_2, x_3 , and so on, which are initially 0 and which can be updated interactively by your

program, with the corresponding update in the printed value of the root of the expression tree.

4. (bonus 35 points) In the ***Towers of Hanoi*** puzzle, we are given a platform with three pegs, a , b , and c , sticking out of it. On peg a is a stack of n disks, each larger than the next, so that the smallest is on the top and the largest is on the bottom. The puzzle is to move all the disks from peg a to peg c , moving one disk at a time, so that we never place a larger disk on top of a smaller one. Describe a recursive algorithm for solving the Towers of Hanoi puzzle for arbitrary n . (Hint: Consider first the subproblem of moving all but the n th disk from peg a to another peg using the third as “temporary storage.”). Then Write a C++ program that can solve instances of the Tower of Hanoi problem.