

Nazara Ibrat

CSCI 313

Problem 1

In the following two text file we are given to comment on, stack linked list and post fix txt. Most of stack linked list was commented on the actual code, it is important to understand how linked implementation of stacks is imputed. This class specifics the basic operation on a stack as a linked list. By using linked lists, we can see how fast data organization will happen, for example an ordered list. If we were to be using an array, it would be an issue to linearly represent and overcome the issue if the size of the array does not meet preconditions. We are using a template class but we need to first we declare the node class. The public class constant linkedStackType is overloading the assignment operator. Each assignment operator has their own function to complete. For example, the void push function is to add a new item to the stack, it has to exist and must not be full or else it will not be used. After the task is completed then the stack has changed and the new item is added to the top of the stack. There is also a private class, which is only a pointer to the stack. It's important to understand linked implementation in stacks because in this question the memory to store the stack elements are allocated dynamically. The point of the question is to understand what the functions are doing. We are introduced to empty stack and full stack. Seeing that we have two new functions, pop and push which regard in the action where the pieces of the stack will be placed.

The second part of the question we had to comment on the program for postfix expressions calculator. We need to understand the operand use encountered in the expression and if it meets all the preconditions. The main algorithm in this code is to initialize the stack, process

the expression, output the result and get the next expression. There we have to use a couple of different function operators to solve this question.