Detailed Report:

This project was interesting. For part A of the project I set up my interfaces. I also set up my student class and my node class. In order to compare the data of the students in my bag I used inheritance. I declared that, my generic variable <T> would inherit from the student class. Doing this would allow me to use the methods in my student class in my bag. For the first part of the project it was very straight forward the issue comes from the second portion of the project. Making the balance method was very easy because we were given the pseudocode. Now when it came to InFix to PostFix that took some time. The pseudocode was not working at first, so I changed the given pseudocode a little bit in order for it to work. Now my method finally works, but one issue with my method is that the operators become doubled, for example ab+ would be ab++. I resolved this issue by doing string.replace(). I replaced all the double operators with a single operator. I still don’t know why my algorithm does that, but I went around the issue by doing string manipulation. Now in order to compare the precedence of each operator, I made an object that takes in an operator and sets its precedence. Now to convert to PreFix I went from Post to Pre. Now the method I used was recursive because I wanted to experiment with recursion. The method for Post to Pre works for all implementations except for the array implementation. I kept getting an error, Other than that, my programs run smoothly.

Output Of Program:

linked List implementation

Freshman: 1

Sophmore: 1

Junior: 2

Senior: 0

resizeable Array Implemntation

Freshman: 1

Sophmore: 1

Junior: 2

Senior: 0

welcome to the Array implementation of the project for the infix

Enter an infix expression:

A \* (B + C) / D

PostFix: abc+\*d/

PreFix: /\*a+bcd

welcome to the LinkedListImplementation of the project for the infix

Enter an infix expression:

A \* (B + C) / D

PostFix: abc+\*d/

PreFix: /\*a+bcd

welcome to the Vector Implementation of the project for the infix

Enter an infix expression:

A \* (B + C) / D

PostFix: abc+\*d/

PreFix: /\*a+bcd