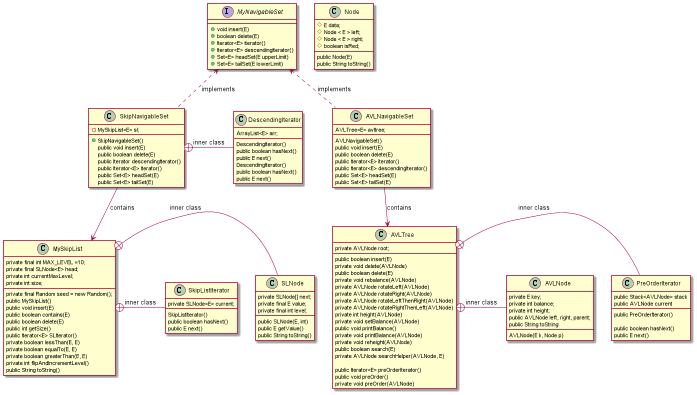
**GTU Department of Computer Engineering**

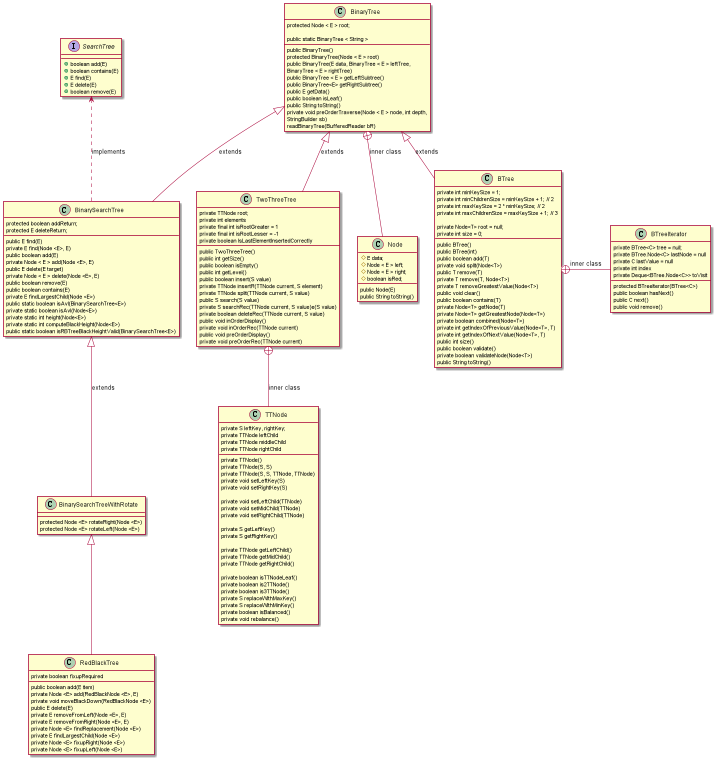
**CSE 222/505 - Spring 2021**

**Homework #7 Report**

**Gökbey Gazi KESKİN**

**1901042631**

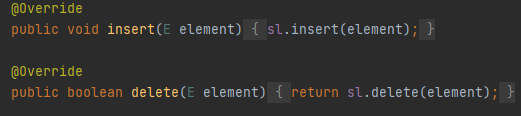
1. **Class Diagrams** *(you can find them as png and puml in the file.)*

****

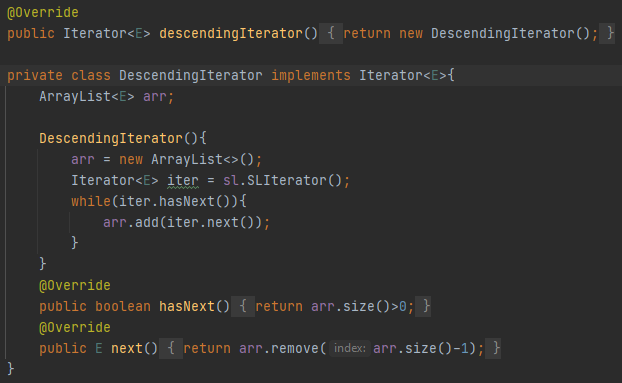
1. **Problem Solution Approach**

**Part 1.1 (SkipNavigableSet)**

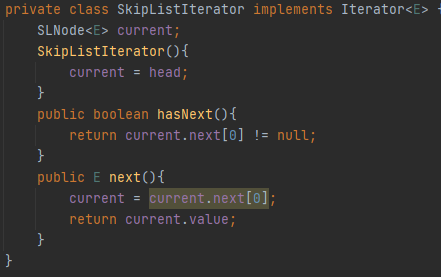
I directly used the insert and delete operations of the skip-list.



For descending iterator, I copied all elements of the skip-list to an ArrayList using forward iterator I implemented. and removed & returned them one by one. Because skip-list only has forwards links.

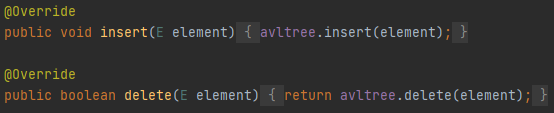


Skip-List Iterator



**Part 1.2(AVLNavigableSet)**

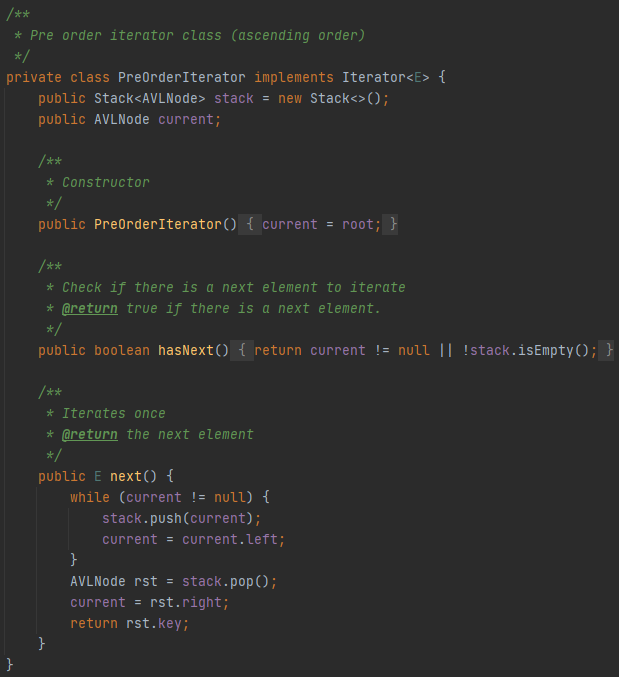
I directly used the insert and delete operations of the AVL tree.



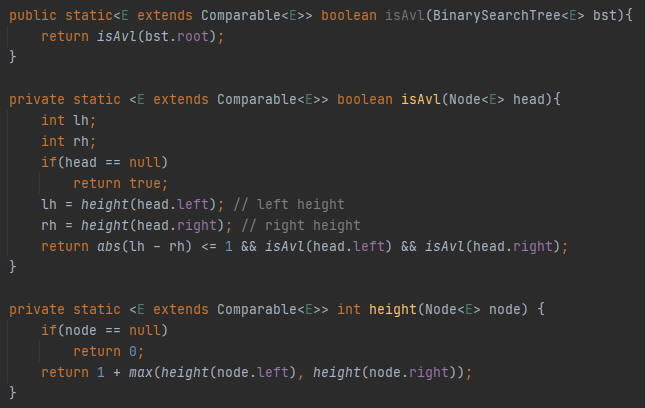
I used the pre order iterator I implemented of the AVL Tree and added them if they are less/more than the given value for head and tail sets.

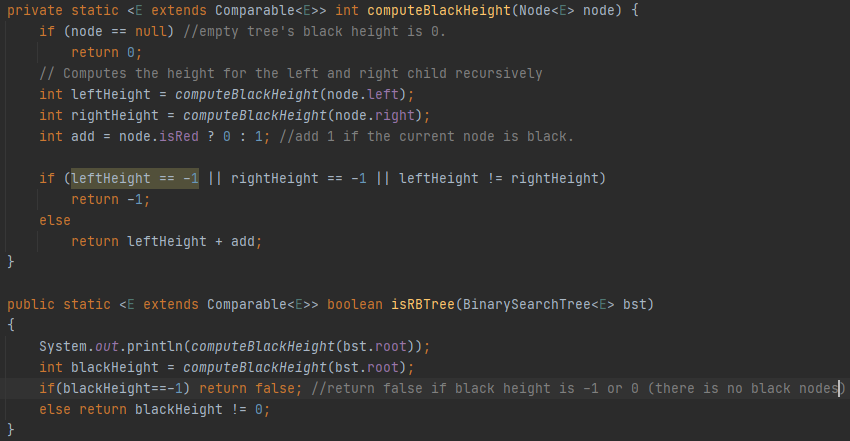


AVL Tree Iterator

****

**Part 2)**

I added methods to BinarySearchTree implementation of the book to determine if the given BST is a AVL Tree / RedBlackTree.

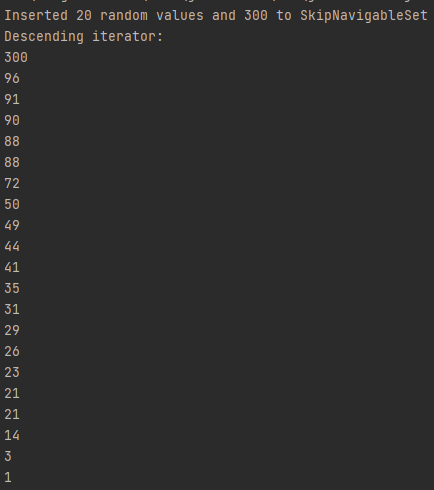
****

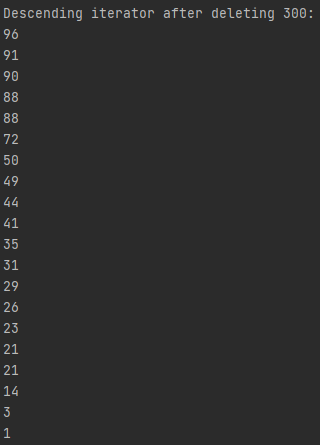
**Part 3)**

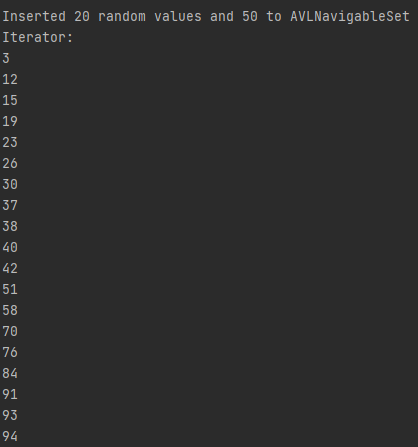
I generated 10 instances for 4 different sizes of given 5 different data structures. (10\*5\*4 = 200 instances in total) Then added 10k 20k 40k 80 to them consecutively. After that, I measured elapsed times for adding 100 extra elements to them using System.nanoTime() and printed them at the end. Results are at test cases.

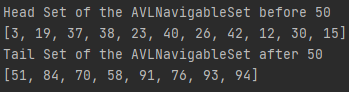
1. **Test Cases & Running Results**

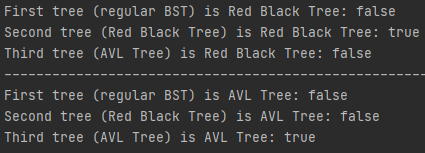
**Part 1.1)**

****

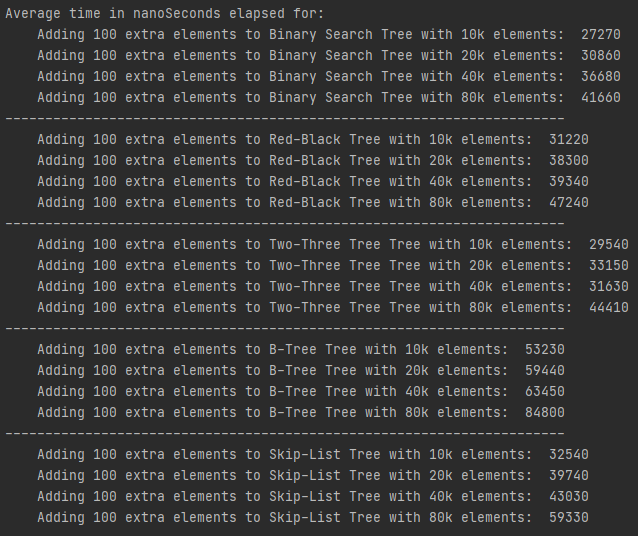
****

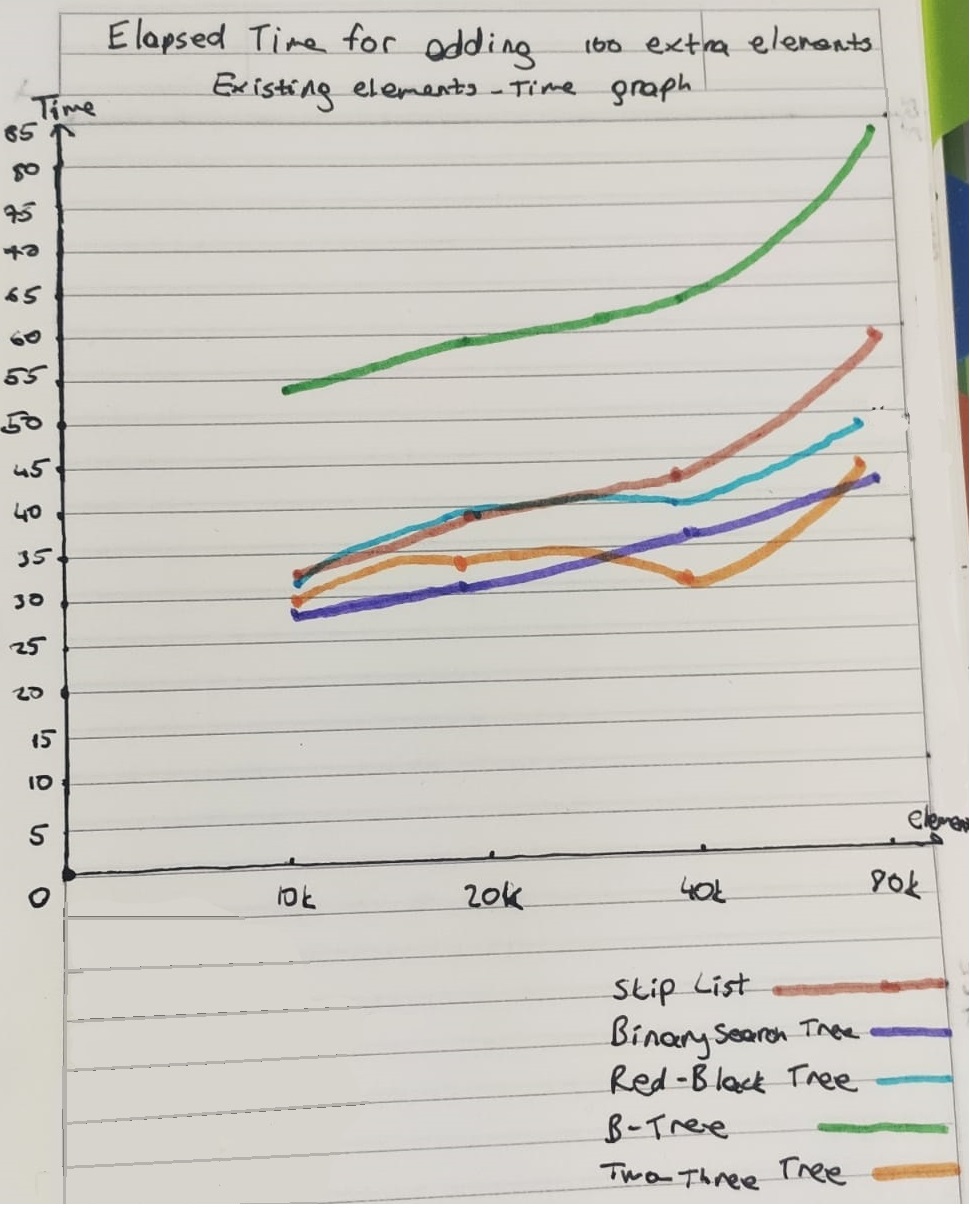
**Part 1.2)**

****

**Part 2)**

**Part 3)**





1. **Detailed System Requirements**

1)Add and remove methods of AVLNavigableSet and SkipNavigableSet needs a generic element as a parameter.





2) headSet and tailSet methods of AVLNavigableSet needs lower-upper bounds as parameter





3)isAvl and isRBTree methods of Binary search tree needs a BST as a parameter to check whether it is a tree of wanted type or not.





Note: I only created JavaDoc for the methods that I wrote.