



In this programming project, you will be implementing Binary Search Tree (BST), AVL tree (AVLTree) and Red-Black tree (RBtree).

BST folder:

You are given:

- Book.java -- No Modifications needed
- BSTDriver.java -- No Modifications needed
- IntObject.java -- No Modifications needed
- SearchTree.java -- No Modifications needed
- Books.txt (data file) -- No Modifications needed
- TestISBN.txt (test file) -- No Modifications needed

You need to read your book pages 315-328 and

- write methods for **size()** and **height()** methods in BinaryTree.java 
- write code for **find()**, **first()** and **last()** methods in BinarySearchTree.java 

You will have six .java files in your *src* folder and two .txt files (data files) in your project folder.

If your code works correctly, your output from the program should match mine. (BSTOutput.txt is attached).



AVLTree Folder:

You cannot start this program unless your BST code is working perfectly.

- Use the SearchTree.java, Book.java, Books.txt, TestISBN.txt, IntObject.java from the BST folder
- BinarySearchTreeWithRotate.java -- You need to complete the code for **rotateLeft** method including the javadoc comments

Use the completed working code for BinarySearchTree.java and BinaryTree.java

AVLTree.java → you need to go over the AVL implementation in your book – pages 477-488 and

- write code for **rebalanceRight()**
- complete the code for the case where **item.compareTo(localRoot.data) > 0** in the add method
- write code for **incrementBalance()** method.

You should have in your *src* folder a total of 8 .java files (and two .txt data files in the project folder):

- | | |
|-----------------------------------|---|
| • Book.java | -- No Modifications needed |
| • AVLDriver.java | -- No Modifications needed |
| • IntObject.java | -- No Modifications needed |
| • SearchTree.java | -- No Modifications needed |
| • Books.txt (data file) | -- No Modifications needed |
| • TestISBN.txt (test file) | -- No Modifications needed |
| • BinarySearchTreeWithRotate.java | -- You need to complete code for rotateLeft() method including javadoc comments |
| • BinarySearchTree.java | -- this code comes from the BST folder |
| • BinaryTree.java | -- this code comes from the BST folder |
| • AVLTree.java | -- you need to write complete working code for rebalanceRight() ; complete the code for the case where item.compareTo(localRoot.data) > 0 in the add method; write the code for incrementBalance() method |

If your code works correctly, your output from the program should match mine. (AVLOutput.txt is attached).

RBTree folder

You cannot start this program unless your AVLTree code is working perfectly.

- Use the SearchTree.java, Book.java, Books.txt, TestISBN.txt, IntObject.java from the BST folder
- Use the completed BinarySearchTreeWithRotate.java from AVLTree folder.
- Use the completed working code for BinarySearchTree.java and BinaryTree.java

RedBlackTree.java → Go over the Red-Black Tree implementation in your book – pages 490-501 and

- complete the code for the case where **item.compareTo(localRoot.data) > 0** in the add method;

You should have in your *src* folder a total of 8 .java files (and two .txt data files in the project folder):

- | | |
|-----------------------------------|--|
| • Book.java | -- No Modifications needed |
| • RBTreeDriver.java | -- No Modifications needed |
| • IntObject.java | -- No Modifications needed |
| • SearchTree.java | -- No Modifications needed |
| • Books.txt (data file) | -- No Modifications needed |
| • TestISBN.txt (test file) | -- No Modifications needed |
| • BinarySearchTreeWithRotate.java | -- This code comes from AVLTree folder |
| • BinarySearchTree.java | -- this code comes from the BST folder |
| • BinaryTree.java | -- this code comes from the BST folder |
| • RedBlackTree.java | -- you need to complete the code for the case where item.compareTo(localRoot.data) > 0 in the add method |

If your code works correctly, your output from the program should match mine.
(RBTreeOutput.txt is attached).

Deliverables:

Your **complete project folder** for BST, AVLTree and RBTree. 

You need to also **write a report summarizing your findings** – you need to address the following:

- Which part of the code or which concept was difficult for you to understand or code and why? And how did you overcome it?
- How many hours in total you spent doing this project?
- Did the output from three programs match your expectation or were there any surprises?
- What did you learn from this programming project?
- Explain when you would use a BST, AVL Tree and RB tree in a real-life application and why?

Due Date: Friday, December 8th 11:59pm

