

WEEK 09

1. PREPARATION FOR ASSIGNMENT

If, and *only if* you can truthfully assert the truthfulness of each comprehension and memory self-check statement below are you ready to start the tasks for the week.

1.1. Reading Comprehension Self-Check.

- I know and understand the difference between an array, a vector, a stack, a queue, a deque, a set, a list, a tree, and a Binary Search Tree.
- I know and understand what a node is and why it is integral to the concept and creation of a tree.
- I know and understand why there is an insert method and no push or pop method for a BST.
- I understand LVR traversal and why it produces a sorted set of data from a BST.
- I understand LVR has to do with the BST iterator.

1.2. **Memory Self-Check.** I can, and have, explained to someone who is not a student in the Computer Science and Electrical Engineering, Computer Information Technology, or Mathematics departments in a way that they understand what a BST is and why it is important.

2. WEEK 09 TEAM TASKS

Note: All review tasks come from the book.

2.1. Review Task 09.3.

2.2. Review Task 09.4.

2.3. Review Task 09.8.

2.4. **Creation Tasks.** Write code matching the UML class diagrams in the git repository so that your implementation passes the assertions in the main.cpp for this data structure.

2.5. **Pondering Task.** What different kinds of uses can you find or brainstorm for this data structure?

Date: January 7, 2020.

2.6. **Pondering Task.** What changes to the API for this data structure can you brainstorm that might make it easier to use? What would be the design for one of your changes?

2.7. **Pondering Task.** Brainstorm some coding tasks/situations or types of data where using this data structure would make organizing, accessing, inserting, or updating the data difficult.

2.8. **Pondering Task.** Brainstorm about and record why there are differences between the API for this data structure and the API for this data structure in another language of your choice. Consider the strengths, weaknesses, similarities, and differences between these API's for the data structure.