**Trees: Implementation and runtime**

To succeed at this challenge, you'll need to demonstrate that you can do the following:

* Implement a binary search tree.

**Instructions**

Your goal for this checkpoint is to get the tests to pass.

To do so, you will be modifying the existing BinarySearchTree class to implement a binary search tree with lookup, insert, and delete capabilities.

**Existing files**

| **File path** | **Description** |
| --- | --- |
| src/main.js | Contains some example uses of the BinarySearchTree class. |
| src/BinarySearchTree.js | Contains the definition of the BinarySearchTree class. The constructor() method has already been completed for you. |

**Tasks**

Complete the following tasks to pass the tests and this assignment.

In the src/BinarySeachTree.js file:

1. Complete the insert() method to insert a node into the BST. The method should accept a key and a value as its arguments.
2. Complete the find() method which should accept a key as its argument and return the value for the given key in the BST. Throw an Error() with a string error message if the node is not found in the BST.
3. Complete the remove() method to delete a node with a given key in the BST. Throw an Error() with a string error message if the node to be removed is not found in the BST.

Once these tasks are complete, all tests should pass.