Problem 2 Testing

1. Test Objective

To verify the correctness of the **O(n) BST Construction** algorithm. This test ensures that:

1. The algorithm correctly reconstructs a Binary Search Tree from a valid preorder traversal array using min/max bounds.
2. The resulting tree structure is valid (verified via Inorder traversal, which must be sorted).
3. The algorithm correctly **detects and rejects** an invalid preorder sequence that cannot form a BST.

---

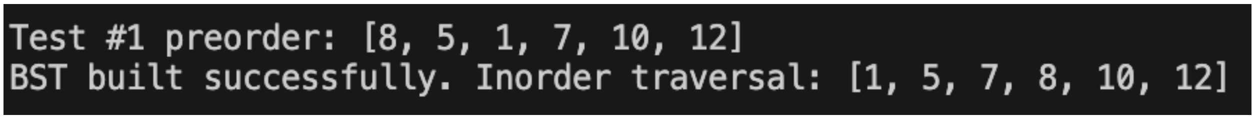
3. Execution Results

Test 1: Valid Preorder Sequence

Description: Run the application. The system processes Test Case #1: `[8, 5, 1, 7, 10, 12]`.

- **Expected Result:**
  - Method returns `true` (Success).
  - Console displays: "BST built successfully."
  - Inorder Traversal output must be sorted: `[1, 5, 7, 8, 10, 12]`.

Evidence:



```
Test #1 preorder: [8, 5, 1, 7, 10, 12]
BST built successfully. Inorder traversal: [1, 5, 7, 8, 10, 12]
```

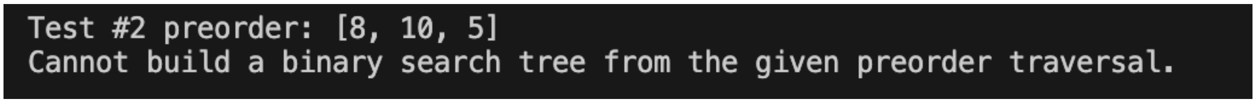
Test 2: Invalid Preorder Sequence

Description: The system processes Test Case #2: `[8, 10, 5]`.

- **Logic:** This is invalid because `5` appears after `10` (which is in the right subtree of `8`). Since `5 < 8`, it cannot exist in the right subtree.
- **Expected Result:**

- Method detects that not all elements could be consumed within valid BST bounds.
- Method returns `false`.
- Console displays: "Cannot build a binary search tree from the given preorder traversal."

Evidence:



```
Test #2 preorder: [8, 10, 5]
Cannot build a binary search tree from the given preorder traversal.
```

4. Conclusion

Overall Status: [PASS]