Tutorial/Lab 11 – Binary Search Trees

Aim

This tutorial/lab aims to explore Binary Search Trees, as well as the associated algorithms such as tree traversals and tree iterators.

Exercise 11.1 Non-recursive Inorder Traversal

- Implement the **nonRecursiveInorder** method in BST using a **stack** instead of recursion.
- And then, write a test program that prompts the user to enter 10 integers, stores them in a BST, and invokes the **nonRecursiveInorder** method to display the elements.

Exercise 11.2 Non-recursive Preorder Traversal

- Implement the **nonRecursivePreorder** method in BST using a **stack** instead of recursion.
- And then, write a test program that prompts the user to enter 10 integers, stores them in a BST, and invokes the **nonRecursivePreorder** method to display the elements.

Exercise 11.3 Find The Leaves

- Define a new class named **BSTWithNumberOfLeaves** that extends BST with the following methods:

```
/** Return the number of leaf nodes */
public int getNumberOfLeaves()
```

Exercise 11.4 Preorder Iterator

- Add the method **preorderIterator** in the BST class that returns an iterator for traversing the elements in a BST in preorder:

```
/** Return an iterator for traversing the elements in preorder */
java.util.Iterator<E> preorderIterator()
```

This is the end of CPT204-2425Tutorial/Lab 11 Task Sheet.

Created Spring 2025
Yihong Wang, Dawei Liu, Erick Purwanto