

CSE 330 LABORATORY -- Week 8, Winter 2020

Instructor: Kerstin Voigt

In this lab, you will be implementing the **Set ADT**. This data structure will be implemented on the basis of Binary Search Trees. In fact, our Set is a Binary Search Tree... On important additions will be by **Set Iterators** which allow us to step through the data structure, accessing one stored value at a time (this is more involved than in-order printing of all values).

Exercise 1: Implement a **class Set<C>** by making few modifications and additions to a **copy of file BinarySearchTree.h**. Consult your lecture (["The Set ADT"](#)) on what to modify and add. The bulk of the additions consists of adding a nested set iterator class, and adding the `begin()` and `end()` member functions into the public interface of class Set.

Test with a straightforward `int main()`: declare a Set suitable to contain integer elements; display the binary tree structure of the Set internals, then use Set iterators to step through the ADT one more time and print out the values in the order in which they are accessed by the iterator.

Exercise 2: Add to your class Set the following member functions:

1. Set Union: returns a new set with is the union of two sets.

```
Set<C> union(Set<C> other) {...}
```

2. Set Intersection: returns the set of elements shared by two sets.

```
Set<T> intersection(Set<C> other) {...}
```

3. Set Difference: returns the set difference between two sets.

```
Set<T> difference(Set<C> other) {...}
```

Test by extending your `int main()` from Exercise 1 as needed: declare a second set with additional values; then demonstrate the computing of union, intersection and difference for both sets.

Credit for this lab (Tuesday Group): (1) Sign up on the signup sheet. (2) Know the the completion of this lab will be written up as Homework Assignment 4.