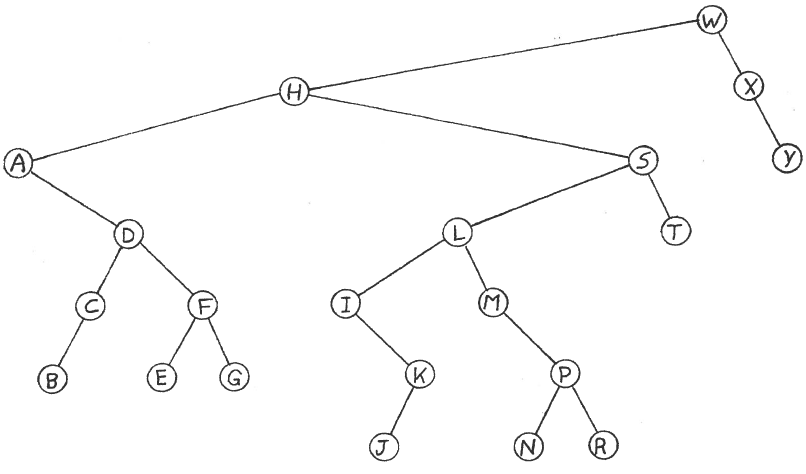
**CSC 1351-3, Spring 2020, Lab 9 Name this project BinarySearchTree**

**Generic Binary Search Tree**

**The binary search tree in today’s lab is shown below:**



**This project will have the following interface and classes:**

**public interface BinarySearchTreeInterface<E extends Comparable>**

This interface is provided in the starter code.

**public class BinarySearchTree<E extends Comparable>**

**implements BinarySearchTreeInterface<E>**

Implement the methods findParent and remove in the starter code.

**public class BinarySearchTreeDemo**

This class is provided in the starter code.

The output of your program should look exactly like the output shown on the next page.

**Output**

**Initial tree:**

**root = W, tree = [A, B, C, D, E, F, G, H, I, J, K, L, M, N, P, R, S, T, W, X, Y]**

**H removed**

**root = W, tree = [A, B, C, D, E, F, G, I, J, K, L, M, N, P, R, S, T, W, X, Y]**

**C removed**

**root = W, tree = [A, B, D, E, F, G, I, J, K, L, M, N, P, R, S, T, W, X, Y]**

**T removed**

**root = W, tree = [A, B, D, E, F, G, I, J, K, L, M, N, P, R, S, W, X, Y]**

**S removed**

**root = W, tree = [A, B, D, E, F, G, I, J, K, L, M, N, P, R, W, X, Y]**

**X removed**

**root = W, tree = [A, B, D, E, F, G, I, J, K, L, M, N, P, R, W, Y]**

**D removed**

**root = W, tree = [A, B, E, F, G, I, J, K, L, M, N, P, R, W, Y]**

**E removed**

**root = W, tree = [A, B, F, G, I, J, K, L, M, N, P, R, W, Y]**

**Y removed**

**root = W, tree = [A, B, F, G, I, J, K, L, M, N, P, R, W]**

**W removed**

**root = I, tree = [A, B, F, G, I, J, K, L, M, N, P, R]**

**L removed**

**root = I, tree = [A, B, F, G, I, J, K, M, N, P, R]**

**I removed**

**root = J, tree = [A, B, F, G, J, K, M, N, P, R]**

**M removed**

**root = J, tree = [A, B, F, G, J, K, N, P, R]**