**Assignment 9 – More on Trees and Sorting**

*Write pseudo-code not Java for problems requiring code. You are responsible for the appropriate level of detail.*

1. **Implement maketree, setleft, and setright for right in-threaded binary trees using the sequential array representation.**
2. **Implement inorder traversal for the right in-thread tree in the previous problem.**
3. **Let’s sort using a method not discussed in class. Suppose you have *n* data values in in array *A*. Declare an array called *Count*. Look at the value in *A[i]*. Count the number of items in *A* that are smaller than the value in *A[i]*. Assign that result to *count[i]*. Declare an output array *Output*. Assign *Output[count[i]] = A[i]*. Think about what the size of *Output* needs to be. Is it *n* or something else? Write a method to sort based on this strategy.**
4. **Analyze the cost of the sort you wrote in the previous problem. What is the impact of random, ordered, or reverse ordered data?**

**3. How many comparisons are necessary to find the largest and smallest of a set of n distinct elements? Do not assume the answer must involve sorting. It could but does not need to do so. Try to be as efficient as you can.**