Type your solutions.

1. Explain carefully, using the definition, why the following statement is meaningless:

“The running time of algorithm A is at least $O(n^2)$.”

2. Prove Theorem 3.1 on page 48:

**Theorem 3.1**

For any two functions $f(n)$ and $g(n)$, we have $f(n) = \Theta(g(n))$ if and only if $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

3. Prove the following using the definitions of $O$, $\Omega$, and $\Theta$.

   (a) $2^{n+1} = O(2^n)$
   (b) $\ln n = \Theta(\log_2 n)$
   (c) $n^\varepsilon = \Omega(\lg n)$ for any $\varepsilon > 0$

4. Read Section B.5 (in Appendix B). Use induction to show that a nonempty binary tree with $n$ nodes has height at least $\lfloor \lg n \rfloor$.

5. Use induction to show that a complete binary tree with height $h$ contains $2^{h+1} - 1$ total nodes.

6. Consider a binary search tree $T$ whose keys are distinct. Prove that if the right subtree of a node $x$ in $T$ is empty and $x$ has a successor $y$, then $y$ is the lowest ancestor of $x$ whose left child is also an ancestor of $x$. (Recall that every node is its own ancestor.)

7. Implement a Binary Search Tree as a template class. Your binary search tree must support the following operations.

   ```cpp
   bool Empty(); // return true if empty; false o/w
   T *Search(T& x); // locate first element with key equal to that in x
   void Insert(T *x); // insert x into the tree
   void Delete(T *x); // delete first element with key equal to that in x
   T *Maximum(); // return the maximum element
   T *Minimum(); // return the minimum element
   T *Successor(T& x); // return the successor of x
   T *Predecessor(T& x); // return the predecessor of x
   void InOrder(); // print elements using an inorder traversal
   void PreOrder(); // print elements using a preorder traversal
   void PostOrder(); // print elements using a postorder traversal
   ```

   $T$ represents the generic type of the data component of each node. Assume that the $T$ type contains a key value and that the comparison operators for $T$ have been overloaded so that they compare the internal key values. Also assume that the stream insertion operator ($\ll$) has been overloaded for $T$.

8. A list can be sorted by inserting the elements into a binary search tree and then extracting them in an in-order traversal.

   (a) What is the worst case running time of this algorithm?
   (b) Implement this sorting algorithm using the template class you wrote above and compare it to the other sorting algorithms you have implemented.