Student Name: _____ Matriculation Number: _____

Instructor: Rongxing Lu
The marking scheme is shown in the left margin and [100] constitutes full marks.

[**60**]   1. Given an integer array $A = \{3, 6, 10, 18, 8, 7, 25, 40\}$,

[20]   (a) Write your own Java source code named `MergeSort.java` to implement Merge Sort algorithm on the array A. Please finish your code in the following template, where "XXXSort" is replaced with "MergeSort".

[20]   (b) Write your own Java source code named `HeapSort.java` to implement Heap Sort algorithm on the array A. Please also finish your code in the following template, where "XXXSort" is replaced with "HeapSort".

[20]   (c) Write your own Java source code named `QuickSort.java` to implement Quick Sort algorithm on the array A. Please also finish your code in the following template, where "XXXSort" is replaced with "QuickSort".
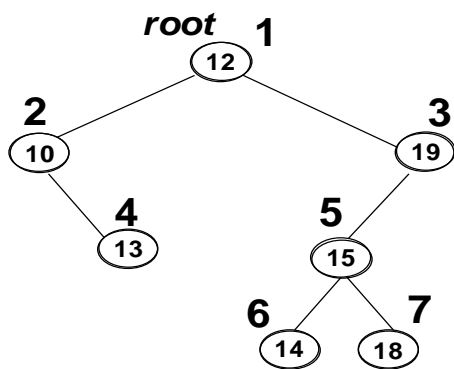
```java
public class XXXSort
{
   public static void main()
   {
     int[] A = {3, 6, 10, 18, 8, 7, 25, 40};
     sort(A);
     show(A);  // display the sorted result A
   }

   public static void sort(int[] A)
   {
      ...
   }

   public static void show(int[] A)
   {
      ...
   }

   ... // add all other functions you need

}
```

[**20**]   2. Write a Java source code named `TreeOrder.java` to first store the following tree (left), where each node is represented by the private class Node (right), and then implement the functions of `PreOrder`, `InOrder`, and `PostOrder` and output the result of each type of order on the tree.

```
private class  Node
{
    int ID;
    int Data;
    int Parent;
    int Left;
    int Right;
}
```

[20]   3. Write a linear-time algorithm in java named `CheckBST.java` to determine whether a binary tree is a binary search tree. Take the following two binary trees as input (For the conveniences, you can first store the two binary trees in your java codes.) to test your algorithm.