**1)**

Apply insertion sort to the following sample array:

Array={6,5,3,11,7,5,2}

Show each step of the algorithm. At each step, you are required to not only create the arrray with its new order, but also explain why you created that sequence.

**2)** What is the time complexity of the following programs? Explain in detail by providing your reasoning.

**a)**

```
function(int n){
    if (n==1)
       return;
    for (int i=1; i<=n; i++){
      for (int j=1; j<=n; j++){
         printf("*");
         break;
      }
    }
}
```

**b)**

```
void function(int n)
    int count = 0;

    for (int i=n/3; i<=n; i++)
       for (int j=1; j+n/3<=n; j = j++)
         for (int k=1; k<=n; k = k * 3)
            count++;
}
```

**3)** We have an unordered array in which we are looking for pairs whose multiplication yields the desired numbers. For example, let our array be {1,2,3,6,5,4} and let the desired number be 6. In this case, our pairs will be (1,6) and (2,3). Provide an algorithm that solves this problem with time complexity O(n(logn)). Express your algorithm as pseudocode (write actual code using Python) and prove that its complexity is O(n(logn)).

**4)** You are given a binary search tree (BST) with $n$ nodes. You are required to merge this tree with another n-node BST. What is the time complexity of this process? Explain in detail by providing your reasoning.

**5)** Suppose that you are given two arrays, where one of them is larger than the other one. Propose a linear time algorithm that finds, if exists, the elements of the small array in the big array. Express your algorithm using pseudocode and calculate the worst case complexity of the algorithm.