**Programming Design, Spring 2015**

**Suggested Solution for Homework 6**

Solution provider: Wei-Hung Liao

**Problem 1**

1.

(0, 6, 3, 5, 2, 7, 4, 2, 5, 6)

(0, 6, 3, 5, 2, 7, 4, 2, 5, 6)

(0, 6, 3, 5, 2, 7, 2, 4, 5, 6)

(0, 6, 3, 5, 2, 2, 4, 5, 6, 7)

(0, 6, 3, 5, 2, 2, 4, 5, 6, 7)

(0, 6, 3, 2, 2, 4, 5, 5, 6, 7)

(0, 6, 2, 2, 3, 4, 5, 5, 6, 7)

(0, 2, 2, 3, 4, 5, 5, 6, 6, 7)

(0, 2, 2, 3, 4, 5, 5, 6, 6, 7)

2.

**Array+1** means adding one unit to the address of the array. In the program, the function recursively calls itself by **array+1** and **n-1**. That is when the function is called to sort some array with length ***n***, it first sorts the last ***n*-1** elements, which form the subarray with length ***n*-1** of original array, then puts the first element to the position it should be.

3.

There are two problems. First, **Array[i]=num1;** should be **array[i-1]=num1;.** If we use **Array[i]=num1;**, the number **array[i]** will disappear. Second, if we write the function like that, the insertion of **num1** won’t happen when the **num1** is the biggest in the subarray. The reason is the function only does the insertion in the **else** section, i.e., when some number is bigger or equal to **num1**.

**Problem 2, Problem 3, and Problem 4**

Please see the .cpp file.