**Bubble Sort**

At the first we have

1. We a public static void method which includes an input that has type of integer array list and it named list;
2. Set needNextPassed = true;

First for loop:

1. For an integer k from 1 to length of the array while we are needing the needNextPassed to be true, and then in each loop k++;
   1. Set needNextPassed = false;

Second for loop:

3.2. For an integer i from 0 to length of the array – k and in each loop, we have i++;

3.2.1. if the if statement true (list[i] > list [i + 1]) then:

3.2.1.1. we create a temporary spot; **int** temp = list[i];

3.2.1.2. then we set the list[i] = list[i+1] the bigger one goes next

3.2.1.3. and then set the list[i+1] = temp;

3.2.1.4. and finally set the needNextPassed = true;

**Insert Sort:**

At the first we have

1. We a public static void method which includes an input that has type of a double array list and it is named list;

Then For loop:

1. For an integer i from 1 to the length of array which is named list; and then in each loop we increase by one (i++);
   1. Set the current element to list[i];

Second for loop:

* 1. For an integer k from (i-1) until k is greater and equal 0 and list[k] is greater than the current element, then in each loop we k--;

2.2.3. we set the list[k]=list[k+1];

2.3. and at the end we have list[k+1] to current element;

**Selection Sort:**

At the first we have:

1. We a public static void method which includes an input that has type of double array list which is named to list;

First for loop:

1. For an integer i from 0 to the length of the array -1 and then increment i by one in each loop;
   1. We set the current min to list[i];
   2. And then we store the location of the current min: **int** currentMinIndex = i;

Second for loop:

* 1. For an integer j from i+1 to length of the array list and then in each loop we increment the j by one j++;
     1. Then we have an if statement that looks over if current min is greater than the unsorted element or not and if so: **if** (currentMin > list[j])
     2. We have current min to list[j];
     3. Then we store the location of the current min index to j;
  2. Then we have another if statement that is used for swapping, which its condition is to check if current min index is not equal to I; currentMinIndex! = i
     1. Then we set the list [current min index] to list[i];
     2. And then list[i] to current min;