

In-Class Activity - Sorting Algorithms

1. Consider the following initial array configuration. In each question, assume we want to sort the array in ascending order (1 is less than 2, 2 is less than 3, etc.).

9	5	7	10	18	1	12	8	16	4
---	---	---	----	----	---	----	---	----	---

- a. Suppose the **insertion** sort algorithm is applied to the **initial array above**. Show the state of the array immediately following **each of the first three** iterations of the outer loop. Please **clearly mark (as in 0th)** the sorted part of the array after each iteration.

0th iteration:

9	5	7	10	18	1	12	8	16	4
---	---	---	----	----	---	----	---	----	---

1st iteration:

--	--	--	--	--	--	--	--	--	--

2nd iteration:

--	--	--	--	--	--	--	--	--	--

3rd iteration:

--	--	--	--	--	--	--	--	--	--

- b. Suppose the **merge** sort algorithm from class is applied to the **initial array above**. Show the state of the two sub-arrays immediately before the final merge.

--	--	--	--	--	--	--	--	--	--

- c. Suppose the **selection** sort algorithm from class is applied to the **initial array above**. In the boxes below, show the state of the array immediately following **each of the first two** iterations of the outer loop. Please **clearly mark** the sorted part and the unsorted part of the array after each iteration. (Initially the sorted part of the array is empty.)

1st iteration:

9	5	7	10	18	1	12	8	16	4
---	---	---	----	----	---	----	---	----	---

2nd iteration:

--	--	--	--	--	--	--	--	--	--

3rd iteration:

--	--	--	--	--	--	--	--	--	--

In-Class Activity - Big-O Performance

Give the Big-O runtime for each of the code snippets below. In each case, n refers to the size of the input list. Answers are worth 2 points each.

a.

```
public static void function1(int[] array) {  
    for(int i = 0; i < 100; i++) {  
        for(int j = 0; j < array.length; j++) {  
            array[j] += i;  
        }  
    }  
}
```

Answer:

b.

```
public static void function2(int[] array) {  
    for(int i = 0; i < array.length; i++) {  
        for(int j = 0; j < i; j++) {  
            if(array[i] != array[j]) {  
                array[j] = array[i];  
            }  
        }  
    }  
}
```

Answer:

c.

```
public static void function3(int[] array) {  
    for(int i = 1; i <= array.length; i++) {  
        for(int j = array.length; j >= 1; j = j / 2) {  
            if(array[i-1] <= array[j-1]) {  
                array[j-1] = array[i-1];  
            }  
        }  
    }  
}
```

Answer: