

✔ Congratulations! You passed!

Grade received 80% To pass 80% or higher

Go to next item

1. Given an array of 6 numbers `-> 6, 8, 19, 48, 9, 90` and applying a **selection sort**. How many swaps must occur before the array is sorted?

1 / 1 point

- ☐ 4
- ☒ 2
- ☐ 6

✔ Correct

That's correct. The array is mostly ordered so only have to swap 19 and 9; and then 48 and 19.

2. Given an array of numbers and a target value, using a loop, what is the worst-case time complexity to check if the number is present in the array?

1 / 1 point

- ☐ `O(1)`
- ☒ `O(n)`
- ☐ `O(log n)`

✔ Correct

That's correct. To determine if a value was there, using a loop would mean checking every element in the array.

3. A binary search can only be performed on a sorted dataset.

1 / 1 point

- ☒ True
- ☐ False

✔ Correct

That's correct. The nature of binary search is that it checks if the value is higher or lower and removes everything beyond the point of that conditional statement.

4. Given the following snippet of pseudocode:

0 / 1 point

```
array = []
```

```
n = 4
```

```
FOR i = 0 TO n:
```

```
    FOR j = 0 TO n:
```

```
        array.add(i*j)
```

What is the space complexity of this problem?

☐ $O(\log n)$

☒ $O(n)$

☐ $O(n^2)$

 **Incorrect**

Not quite. This would indicate that the space used would reflect the size of the input.

5. What advantage is there to changing element location using an in-place swap?

1 / 1 point

☐ It is a memory feature that allows many variables to reference the same memory location.

☐ It reduces the time taken to complete an algorithm through lowering the time complexity.

☒ It reduces the amount of space taken by removing the need to create another variable in memory.

 **Correct**

That's correct. In-place swapping is done to arrays in place of creating new ones and storing the sorted data there. It is a good process for reducing the space complexity of a solution.