Coding the sorting algorithms

General Notes:

You are only required to complete one of the 2 questions. But feel free to complete both if you are interested.

Please leverage the test cases we provided at the bottom of each of the code files to debug your program. Those test cases cover different scenarios of the given tasks, and provide a way for you to check answers.

If some of the test cases are failing, that means you have some obvious scenarios that you are not covering yet. Try to dry run the code with the failing test cases with your mind or a piece of paper, and find out what you are missing.

1.1 In this part, you are going to try coding the Bubble Sort algorithm we discussed in class.

Open `bubble-sort.js` file and please finish the `bubbleSort` function according to the specification in the file. We have prepared some test cases for you to test your functions.

1.2 In this part, you are going to try coding the “Find Substring” algorithm question we discussed in class

Open `insertion.js` file and please finish the `insertionSort` function according to the specification in the file. We have prepared some test cases for you to test your functions.

Lecture 5: Sorting Algorithms - Part 1

Agenda:

- Sorting Algorithms
- Bubble Sort
- Insertion Sort

Sorting Algorithms

Sorting Algorithms

Sorting refers to sorting making a randomly ordered array into an array with increasing or decreasing order.

27, 34, 10, 70, 48

↓ Sorting in an increasing order

10, 27, 34, 48, 70

Source: https://lamfo-unb.github.io/2019/04/21/Sorting-algorithms/