Instructions: In-class exercises are meant to introduce you to a new topic and provide some practice with the new topic. **Work in a team of up to 4 people to complete this exercise.** You can work simultaneously on the problems, or work separate and then check your answers with each other. **Turn in one copy of the exercise per group**.

Names:

Algorithms: Examples of Algorithms

Algorithms
In Computer Science, searching and sorting algorithms are common to come across, since data always needs to be accessed and organized.

Linear Search

A Linear Search is a simple searching algorithm that starts at the beginning of a list and goes forward one at a time, inspecting each element of the list for some data. It is not terribly efficient, but it is easy to implement.

```
1  #INPUTS:  searchList: A list of values to search through
2  #         findMe: The value to find in the list
3  #OUTPUTS: -1: Returned when the findMe item is not found
4  #         index: The index where findMe is found
5
6  def LinearSearch( searchList, findMe ):
7      # Search every item in the list
8      for i in range( 0, len( searchList ) ):
9          if ( searchList[i] == findMe ):
10             return i    # Found at this index
11
12     return -1 # Not found
```


By Rachel Singh, last updated October 17, 2018

Question 1
Step through the LinearSearch function on the previous page, writing down the variable values at each step.

Function call: `LinearSearch( [ 1, 9, 3, 8 ], 5 )`

Line	findMe	i	searchList[i]	Action
6	5	i	-	Enter function

Insertion Sort: [1]

```
1 # INPUT: seq a list of numbers
2 # OUTPUT: seq is sorted after this function runs
3
4 def InsertionSort( seq ):
5     for i in range(1, len(seq)):
6         j = i
7         while j > 0 and seq[j - 1] > seq[j]:
8             seq[j - 1], seq[j] = seq[j], seq[j - 1]
9             j -= 1
```

[1] From https://stackoverflow.com/questions/12755568/how-does-python-insertion-sort-work#28727652

Question 2
Step through the InsertionSort function on the previous page, writing down the variable values at each step.

Function call: `InsertionSort( [ 2, 5, 3, 1 ] )`

Line	seq	i	j	seq[j-1]	seq[j]	Action
4	[2,5,3,1,6]	-	-	-	-	Enter function