Nelson Dong
Hw01
1. An implementation being correct is an optimally written implementation that takes inputs from an appropriate domain as parameters and creates an outcome that satisfies the conditions asked of the implementation and avoids any restrictions.
2. We can prove the correctness of this implementation using induction.
    a. Isort:
        i. This isort is optimally written with no extra, unnecessary "code".
        ii. The inputs are correct with the given domain, a list of numbers.
        iii. The output is another list of numbers, which satisfies the conditions of list N → list N
        iv. Induction :
            1. Base case: an empty list with no elements or 1 element is already sorted
            2. Inductive hypothesis : using Insrt on an element and an ordered list will return another ordered list with that element sorted into that list
            3. Inductive step: suppose a list of n elements. The first element x is unordered from the rest of the list, which is size n -1. Then by calling insrt with that element x with the sorted list size n-1, we will get a new list of size n according to the inductive hypothesis. If the rest of the list is unsorted, we call the isort algorithm on the list of size n-1. If the list reaches a size of 1 or 0, it is sorted. Therefore, if a list of size n is called with isort, the list will become sorted.
3. The definition is complete because any buggy implementations would not satisfy my definition of correct.
    a. One buggy implementation would be a list with elements that are not numbers or contain both numeric and non-numeric elements.
        i. This is buggy because the implementation cannot compare whether a number is greater than a letter
        ii. This would not pass my definition of correctness because it fails the condition of an appropriate domain.
    b. Another buggy implementation would be having the algorithm return an object that is not a list of elements.
        i. This is buggy because the insrt takes a list of numbers, so having insrt return something that is not a list will break the algorithm
        ii. This would not pass my definition because it would not be an appropriate domain for the insrt and it is not an outcome that satisfies the conditions needed for it.