Lab 3 - Software Unit Testing with PyTest

SORT\_ASCENDING = 0

SORT\_DESCENDING = 1

def bubble\_sort(arr, sorting\_order):

# Copy input list to results list

arr\_result = arr.copy()

# Get number of elements in the list

n = len(arr\_result)

if n < 10:

# Traverse through all array elements

for i in range(n - 1):

# range(n) also work but outer loop will

# repeat one time more than needed.

# Last i elements are already in place

for j in range(0, n - i - 1):

if sorting\_order == SORT\_ASCENDING:

if arr\_result[j] > arr\_result[j + 1]:

arr\_result[j], arr\_result[j + 1] = arr\_result[j + 1], arr\_result[j]

elif sorting\_order == SORT\_DESCENDING:

if arr\_result[j] < arr\_result[j + 1]:

arr\_result[j], arr\_result[j + 1] = arr\_result[j + 1], arr\_result[j]

else:

# Return an empty array

arr\_result = []

else:

arr\_result = -1

return arr\_result

def main():

# Driver code to test above

arr = [64, 34, 25, 12, 22, 11, 90]

# Sort in ascending order

result = bubble\_sort(arr, SORT\_ASCENDING)

print("\nSorted array in ascending order: ")

print(result)

# Sort in descending order

print("Sorted array in ascending order: ")

result = bubble\_sort(arr, SORT\_DESCENDING)

print(result)

if \_\_name\_\_ == "\_\_main\_\_":

main()

Pytest

import Lab3

print("Test\_Lab3")

def test\_bubble\_sort\_ascending():

result = []

input\_arr = [64, 34, 25, 12, 22, 11, 90]

test\_arr = [11, 12, 22, 25, 34, 64, 90]

result = Lab3.bubble\_sort(input\_arr, Lab3.SORT\_ASCENDING)

assert (result == test\_arr)

def test\_bubble\_sort\_descending():

result = []

input\_arr = [64, 34, 25, 12, 22, 11, 90]

test\_arr = [90, 64, 34, 25, 22, 12, 11]

result = Lab3.bubble\_sort(input\_arr, Lab3.SORT\_DESCENDING)

assert (result == test\_arr)

def test\_bubble\_sort\_invalid():

result = []

input\_arr = [64, 34, 25, 12, 22, 11, 90]

result = Lab3.bubble\_sort(input\_arr, 3)

assert (result == [])