hw5 az147

```
"""
SelectionSort
"""
def acceptance_test(listToTest):
    if listToTest == []:
        return True
    else:
        for index in range(len(listToTest) - 1):
            if(listToTest[index] > listToTest[index + 1]):
                return False
        return True
def SelectionSort(listToSort):
    for index in range(len(listToSort)):
        min = listToSort[index]
        minindex = index
        for index1 in range(index, len(listToSort)):
            #bug inserted here
            if min > listToSort[index1]:
                min = listToSort[index1]
                minindex = index1
        temp = listToSort[index]
        listToSort[index] = listToSort[minindex]
        listToSort[minindex] = temp
    return listToSort

"""
InsertionSort
"""
def InsertionSort(listToSort):
    isswap = 0
    for index in range(len(listToSort)):
        isswap = 0
        for index1 in range(index, 0, -1):
            if listToSort[index1] < listToSort[index1 - 1]:
                temp = listToSort[index1]
                listToSort[index1] = listToSort[index1 - 1]
                listToSort[index1 - 1] = temp
                isswap = 1
            if isswap == 0:
                break
            isswap = 0
    return listToSort

if __name__ == "__main__":
    input_string = input("Enter a list element separated by space ")
    listToSort = input_string.split()
    for num in range(len(listToSort)):
        listToSort[num] = int(listToSort[num])
    print("calling selection sort")
    result = SelectionSort(listToSort)
    t = acceptance_test(result)
    if t == False:
```

```
        print('first block(slection sort) is wrong! now try second block(insertion sort)')
        result = InsertionSort(listToSort)
        t = acceptance_test(result)
        if t==False:
            raise Exception('Can not get the right sorting result!')
    print('the right sorting result is')
    print(result)
```

this code use SelectionSort as the first block and the SelectionSort as the second block

and the acceptance_test function is for checking is the result from a block of code is right.

1. if the first block works right

```
oliverrr@Aohuas-MagicBook hw5 % /usr/bin/python3 "/Users/oliverrr/Desktop/E
Enter a list element separated by space 1 4 4 2 7 4 8 24 5424 12 45243 121
calling selection sort
the right sorting result is
[1, 2, 4, 4, 4, 7, 8, 12, 24, 121, 5424, 45243]
```

the first block(selection sort) is called and the result is checked and printed out

2. if the first block gives the wrong result(manually insert bug in the first block), and the second block works right

```
oliverrr@Aohuas-MagicBook hw5 % /usr/bin/python3 "/Users/oliverrr/Desktop/ECE59
Enter a list element separated by space 1 4 4 2 7 4 8 24 5424 12 45243 121
calling selection sort
first block(slection sort) is wrong! now try second block(insertion sort)
the right sorting result is
[1, 2, 4, 4, 4, 7, 8, 12, 24, 121, 5424, 45243]
```

the second block will be called as the first result failed the acceptance test, and after the second block give the right result. We print out the right result.

3. if both of the blocks can not give us the right result

```
oliverrr@Aohuas-MagicBook hw5 % /usr/bin/python3 "/Users/oliverrr/Desktop/ECE590 software reliabili
Enter a list element separated by space 1 2 5 123 85 123 465636 123 234 123
calling selection sort
first block(slection sort) is wrong! now try second block(insertion sort)
Traceback (most recent call last):
  File "/Users/oliverrr/Desktop/ECE590 software reliability/hw5/hw5/hw5.py", line 57, in <module>
    raise Exception('Can not get the right sorting result!')
Exception: Can not get the right sorting result!
```

the program will throw an exception accordingly and exit.