Midterm Lab Task 3 - Python List Collections

Problem 1. Using List Collection type. Create a program that will allow the user to perform the following **functions:** (add, update, search, delete, display, and sort) items in a list:

Note: You are free to decide what data you will be storing in the list and name the list based on the type of data you wish to store.

[MENU OPTIONS]

1 – Add Items
2 – Search for an Item
3 – Remove an Item
4 – View all items (Sorted either A-Z | Z -A)
0 – Exit program

Pick one [0 to quit]: ___

Requirements:

1. The user can add items in the list until the user presses x to stop
2. The user should be able to perform **search** if an item exists – Display if found or not found and count the number of instance in the list.
3. The user should also be given the option to remove an item in the list – Display the Message "Item found and deleted" once deletion is performed – else display "item not found-deletion unsuccessful"
4. The user may also opt to view items in the list and display items sorted in Ascending order
5. The user may opt to exit the program by typing 0

Note: you are free to design the interface of the program, base on the Menu options shown.

Angeles, Gabriel Elmo L.
BSCS - C204

```
computer_parts = []

def add_items():
    while True:
        item = input("Enter computer part to add (x to stop): ")
        if item.lower() == "x":
            break
        computer_parts.append(item)
    print("Items added successfully.\n")

def search_item():
    item = input("Enter computer part to search: ")
    if item in computer_parts:
        print(item, "found in the list.")
        print("Number of instances:", computer_parts.count(item))
    else:
        print(item, "not found in the list.")

def remove_item():
    item = input("Enter computer part to remove: ")
    if item in computer_parts:
        computer_parts.remove(item)
        print(item, "found and deleted.")
    else:
        print(item, "not found - deletion unsuccessful.")

def view_items():
    if len(computer_parts) == 0:
        print("The list is empty.")
    else:
        order = input("Sort A-Z or Z-A? (A/Z): ")
        if order.upper() == "A":
            sorted_list = sorted(computer_parts)
        else:
            sorted_list = sorted(computer_parts, reverse=True)
        print("Computer Parts in the list:")
        for i in sorted_list:
            print(i)

while True:
    print("\n[ MENU OPTIONS ]")
```

Sample Output:

```
[ MENU OPTIONS ]
1 - Add Computer Parts
2 - Search for a Computer Part
3 - Remove a Computer Part
4 - View all Computer Parts (Sorted)
0 - Exit program
Pick one [0 to quit]: 1
Enter computer part to add (x to stop): Motherboard
Enter computer part to add (x to stop): CPU
Enter computer part to add (x to stop): RAM
Enter computer part to add (x to stop): GPU
Enter computer part to add (x to stop): x
Items added successfully.
```

```
[ MENU OPTIONS ]
1 - Add Computer Parts
2 - Search for a Computer Part
3 - Remove a Computer Part
4 - View all Computer Parts (Sorted)
0 - Exit program
Pick one [0 to quit]: 2
Enter computer part to search: CPU
CPU found in the list.
Number of instances: 1
```

```
[ MENU OPTIONS ]
1 - Add Computer Parts
2 - Search for a Computer Part
3 - Remove a Computer Part
4 - View all Computer Parts (Sorted)
0 - Exit program
Pick one [0 to quit]: 3
Enter computer part to remove: Motherboard
Motherboard found and deleted.
```

```
print("1 - Add Computer Parts")
print("2 - Search for a Computer Part")
print("3 - Remove a Computer Part")
print("4 - View all Computer Parts (Sorted)")
print("0 - Exit program")

choice = input("Pick one [0 to quit]: ")

if choice == "1":
    add_items()
elif choice == "2":
    search_item()
elif choice == "3":
    remove_item()
elif choice == "4":
    view_items()
elif choice == "0":
    print("Exiting program...")
    break
else:
    print("Invalid choice. Try again.")
```