Midterm Lab Task 3
List Collections

Problem 1. Using the 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:

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)
0 – Exit program

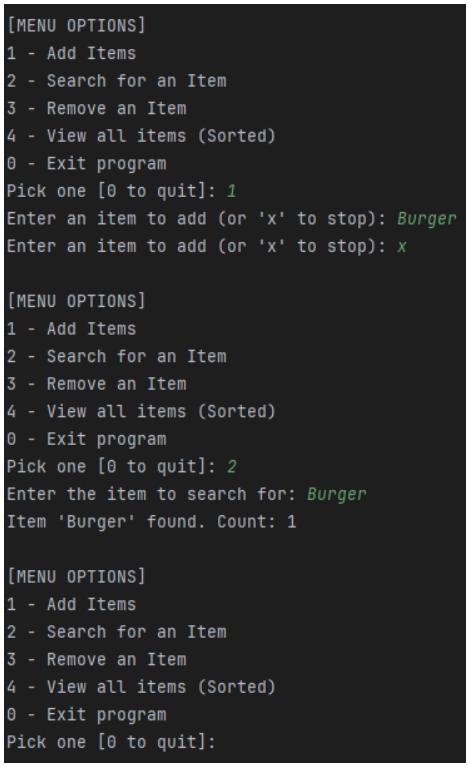
Pick one [0 to quit]: ___
Requirements:

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

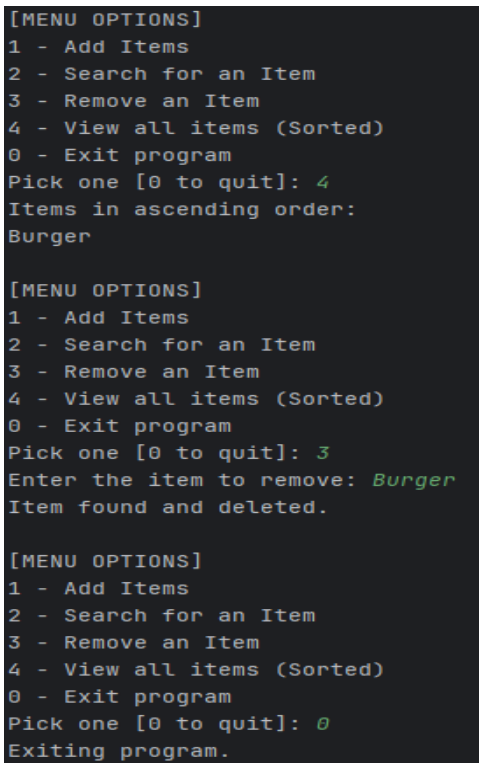
Note: you are free to design the interface of the program, base on the Menu options shown.

```python
items = []

def add():  1usage
    while True:
        item = input("Enter an item to add (or 'x' to stop): ")
        if item.lower() == 'x':
            break
        items.append(item)

def search():  1usage
    search_term = input("Enter the item to search for: ")
    count = 0
    found = False
    for item in items:
        if item == search_term:
            found = True
            count += 1
    if found:
        print(f"Item '{search_term}' found. Count: {count}")
    else:
        print(f"Item '{search_term}' not found.")

def remove():  1usage
    remove_term = input("Enter the item to remove: ")
    if remove_term in items:
        items.remove(remove_term)
        print("Item found and deleted.")
    else:
        print("Item not found - deletion unsuccessful.")

def view():  1usage
    if not items:
        print("The list is empty.")
    else:
        sorted_items = sorted(items)
        print("Items in ascending order:")
        for item in sorted_items:
            print(item)

while True:
    print("\n[MENU OPTIONS]")
    print("1 - Add Items")
    print("2 - Search for an Item")
    print("3 - Remove an Item")
    print("4 - View all items (Sorted)")
    print("0 - Exit program")

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

    if choice == '1':
        add()
    elif choice == '2':
        search()
    elif choice == '3':
        remove()
    elif choice == '4':
        view()
    elif choice == '0':
        print("Exiting program.")
        break
    else:
        print("Invalid choice. Please try again.")
```

Sample Output



```
[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]: 1
Enter an item to add (or 'x' to stop): Burger
Enter an item to add (or 'x' to stop): x

[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]: 2
Enter the item to search for: Burger
Item 'Burger' found. Count: 1

[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]:
```



```
[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]: 4
Items in ascending order:
Burger

[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]: 3
Enter the item to remove: Burger
Item found and deleted.

[MENU OPTIONS]
1 - Add Items
2 - Search for an Item
3 - Remove an Item
4 - View all items (Sorted)
0 - Exit program
Pick one [0 to quit]: 0
Exiting program.
```