//Assignment 7

Title: Sort Online Orders by Delivery Time using Merge Sort. Problem Statement: You are given a list of online orders, each with an estimated delivery time in minutes. Write a program to sort these orders using the Merge Sort algorithm so the delivery system can prioritize quicker deliveries first.

```
def merge(arr, left, mid, right):
    """Merge two sorted halves of arr."""
    # Create temporary subarrays
    L = arr[left:mid+1]
    R = arr[mid+1:right+1]
    i = j = 0
    k = left
    # Merge while comparing delivery times
    while i < len(L) and j < len(R):
        if L[i][1] <= R[j][1]:  # Compare delivery times
            arr[k] = L[i]
            i += 1
        else:
            arr[k] = R[j]
            j += 1
        k += 1
    # Copy remaining elements
    while i < len(L):
        arr[k] = L[i]
        i += 1
        k += 1
```

```
    while j < len(R):
        arr[k] = R[j]
        j += 1
        k += 1
def merge_sort(arr, left, right):
    """Recursive merge sort."""
    if left < right:
        mid = (left + right) // 2
        merge_sort(arr, left, mid)
        merge_sort(arr, mid+1, right)
        merge(arr, left, mid, right)
def display_orders(orders):
    """Pretty print the orders."""
    print("\nOrders:")
    print("{:<5} {:<20} {:<15}".format("ID", "Item", "Delivery Time (min)"))
    for order in orders:
        print("{:<5} {:<20} {:<15}".format(order[0], order[1], order[2]))
def main():
    orders = []
    while True:
        print("\n--- Online Orders Merge Sort Menu ---")
        print("1. Add Order")
        print("2. Display Orders")
        print("3. Sort Orders by Delivery Time")
        print("4. Exit")
        choice = input("Enter choice: ")
```

```
        if choice == "1":
            order_id = input("Enter Order ID: ")
            item_name = input("Enter Item Name: ")
            delivery_time = int(input("Enter Delivery Time (in minutes): "))
            orders.append((order_id, item_name, delivery_time))
        elif choice == "2":
            if orders:
                display_orders(orders)
            else:
                print("No orders available!")
        elif choice == "3":
            if orders:
                merge_sort(orders, 0, len(orders)-1)
                print("\nOrders sorted by delivery time:")
                display_orders(orders)
            else:
                print("No orders to sort!")
        elif choice == "4":
            print("Exiting program...")
            break
        else:
            print("Invalid choice! Please try again.")


if __name__ == "__main__":
    main()
```

//output

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: 1

Enter Order ID: 101

Enter Item Name: laptop

Enter Delivery Time (in minutes): 40

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: 1

Enter Order ID: 102

Enter Item Name: mobile

Enter Delivery Time (in minutes): 45

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: 1

Enter Order ID: 103

Enter Item Name: book

Enter Delivery Time (in minutes): 50

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: 2

Orders:

ID	Item	Delivery Time (min)
101	laptop	40
102	mobile	45
103	book	50

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: 3

Orders sorted by delivery time:

Orders:

ID	Item	Delivery Time (min)
103	book	50
101	laptop	40
102	mobile	45

--- Online Orders Merge Sort Menu ---

1. Add Order

2. Display Orders

3. Sort Orders by Delivery Time

4. Exit

Enter choice: