NAME:S.ROSHINI

ROLL NUMBER:24UCS125

class Node:

    def \_\_init\_\_(self, data):

        self.data = data

        self.next = None

class Queue:

    def \_\_init\_\_(self):

        self.front = None

        self.rear = None

    def enqueue(self, value):

        new\_node = Node(value)

        if self.rear is None:  # Queue is empty

            self.front = self.rear = new\_node

        else:

            self.rear.next = new\_node

            self.rear = new\_node

        print(f"{value} enqueued to queue.")

    deQUE(self):

        if self.front is None:

            print("Queue is EMPTY! Cannot dequeue.")

        else:

            removed = self.front.data

            self.front = self.front.next

            if self.front is None:  # If queue becomes empty

                self.rear = None

            print(f"{removed} dequeued from queue.")

    def display(self):

        if self.front is None:

            print("Queue is EMPTY!")

        else:

            print("Queue elements are:")

            temp = self.front

            while temp is not None:

                print(f"{temp.data} --> ", end="")

                temp = temp.next

            print("NULL")

queue = Queue()

while True:

    print("\n--- Linked List Queue Menu ---")

    print("1. Enqueue")

    print("2. Dequeue")

    print("3. Display")

    print("4. Exit")

    choice = input("Enter your choice (1-4): ")

    if choice == '1':

        value = input("Enter value to enqueue: ")

        queue.enqueue(value)

    elif choice == '2':

        queue.dequeue()

    elif choice == '3':

        queue.display()

    elif choice == '4':

        print("Exiting program. Goodbye!")

        break

    else:

        print("Invalid choice. Please try again.")