Class Structure:

- ◆ **Node Class**
  - Represents one queue element.
  - Stores:
    - Data: value of the node.
    - next: pointer to the next node.
- ◆ **Queue Class**
  - Uses two pointers:
    - head: front of the queue (for pop)
    - tail: rear of the queue (for push)

Function Logic:

1. empty()

- Checks if the queue is empty.
- Returns true if head == NULL.

2. push(int x) → Enqueue

- Creates a new node with value x.
- If queue is empty: head = tail = newNode.
- Else: attach newNode at tail, then update tail.

Adds value at the rear (end) of the queue.

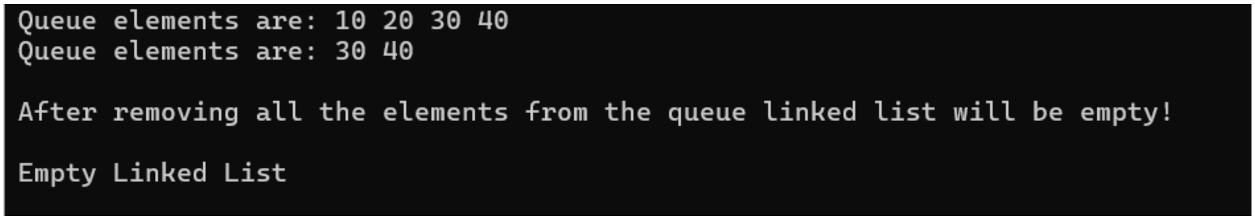
3. pop() → Dequeue

- If queue is empty: show a message.
- Else: remove the front node (head) and move head to the next node.

Removes the value from the front of the queue.

4. Display()

- If empty: print message.
- Else: loop from head to NULL, print each Data.



```
Queue elements are: 10 20 30 40
Queue elements are: 30 40

After removing all the elements from the queue linked list will be empty!

Empty Linked List
```