==

| EMERGENCY ROOM QUEUE MANAGEMENT |
| Using Doubly Linked List |

==

Student Name:Muhammad Anas Afridi Bs Ai group b

----

1. PROBLEM STATEMENT
A hospital ER needs a flexible queue for patients:
- Critical: Add to front
- Walk-in: Add to end
- Nurse priority: Insert at exact position k (1-based)
- Treat: Remove from front

Handles dynamic changes with Doubly Linked List.

----

2. PROPOSED SOLUTION
- Node: patientID, prev*, next*
- Operations: insertAtBeginning(), insertAtEnd(), insertAtPosition(k), deleteFromBeginning()
- Edge Cases: Empty list, single node, pos=1, pos > length (→ end)

(See C++ Code on GitHub)

----

3. GRAPHICAL REPRESENTATION (Dry Run)

1. insertAtEnd(101)
   NULL ← [101] → NULL   (H/T)

2. insertAtEnd(102)
   [101] ↔ [102]   (H →   → T)

3. insertAtBeginning(200)
   [200] ↔ [101] ↔ [102] (H →   → T)

4. insertAtPosition(150, 2)
   [200] ↔ [150] ↔ [101] ↔ [102]

5. deleteFromBeginning()
   [150] ↔ [101] ↔ [102] (H →   → T)

6. insertAtEnd(300) **FINAL**
   [150] ↔ [101] ↔ [102] ↔ [300]
   Head=150 | Tail=300
   Forward: 150↔101↔102↔300
   Backward:300↔102↔101↔150

-GitHub Repo:
- Complete C++ Code
- Poster PNG
- Poster PDF

Thank You!

==