# CL2001 – Data Structure Lab

# Assignment # 1

**Note:**

- Submit a pdf file containing all your C++ code with all possible screenshots of every task output on Google Classroom.
- **Also submit .cpp extension file with pdf.**
- Copied task will be awarded **zero** marks.
- Submit your file in this format (roll-no-name) i.e (23P-1234-Ali.pdf).

## Problem: 1

The **Josephus problem** is a famous theoretical problem in computer science and mathematics. It goes like this: there are 'n' people standing in a circle waiting to be executed. Starting from a specified position in the circle, you need to count m people clockwise or anticlockwise (depending on the variation) and eliminate that person. The process is repeated with the remaining people until only one person remains, who is then declared the '**survivor**'.

Create a C++ program that presents users with a menu-based interface to simulate the Josephus problem. The program should allow users to dynamically add people to the circle and then initiate the elimination process from a designated starting point, following a specified jump value and direction.

Here's a breakdown of the requirements:

1. **Menu Options**:
    - Add Person: Allows users to add person to the circle dynamically, assigning labels to each person.
    - Start Elimination Process: Initiates the elimination process, prompting the user to specify the starting point, jump value, and direction (clockwise or anticlockwise).
    - Exit: Terminates the program.
2. **Josephus Problem Simulation**:
    - Users can add persons to the circle at runtime through the 'Add Person' option.
    - Once the circle is set up, users can choose the 'Start Elimination Process' option to begin the elimination.
    - The program should display the sequence of eliminations and declare the survivor at the end.
3. **Data Structure**:
    - Implement a circular doubly linked list to represent the circle of persons.
    - Each person should be represented as a node in the list, containing information such as their label.
4. **User Interaction**:
    - The program should provide clear prompts and instructions to guide users through the menu options and input requirements.
    - After each operation, the program should return to the main menu to allow users to choose their next action.

5. **Sample Output**:
    - Provide sample output demonstrating the functionality of the program, including adding persons, initiating the elimination process, and displaying the result.

Your task is to design and implement this menu-based program in C++, ensuring it follows the specified requirements and provides an intuitive user experience. Additionally, include comments in your code to explain key functionalities and provide clarity.

**Sample Output:**

**--- Josephus Problem Simulator ---**

1. Add Person
2. Start Elimination Process
3. Exit

Enter your choice: 1
Enter the name for the person: John
Person John added to the circle.

Current persons in the circle: John

1. Add Person
2. Start Elimination Process
3. Exit

Enter your choice: 1
Enter the name for the person: Mary
Person Mary added to the circle.

Current persons in the circle: John, Mary

1. Add Person
2. Start Elimination Process
3. Exit

Enter your choice: 1
Enter the name for the person: Nick
Person Nick added to the circle.

Current persons in the circle: John, Mary, Nick

1. Add Person
2. Start Elimination Process
3. Exit

Enter your choice: 2
Enter the starting point: Mary
Enter the jump value: 2
Choose the direction (1. Clockwise / 2. Anticlockwise): 1

Elimination sequence:

1. Eliminate person John
2. Eliminate person Mary

The survivor is Nick.

1. Add Person
2. Start Elimination Process
3. Exit

Enter your choice: 3

Exiting the program...

---