

# EXP-6: Implementation of Cyclic Redundancy Check (CRC)

## **I. Aim:**

Implement Error Detection technique using any Programming Language.

## **II. Components and Tools:**

System: Desktop Computer/Laptop

Operating system: Windows/Linux

Language: any language

## **III. Description ( Problem Statement):**

## **IV. Algorithm:**

## **V. Program:**

## **VI. Output Description/Analysis:**

## **VII. Viva-Voce Questions:**

# Description ( Problem Statement)

## Cyclic Redundancy Check (CRC)

In digital communication systems, the Cyclic Redundancy Check (CRC) is a widely used error-detection technique that helps ensure the integrity of transmitted data. It involves appending a checksum (remainder) to the data before transmission and verifying it at the receiver's end to detect any errors that might have occurred during transmission. Your task is to implement a CRC-based error detection system for a given dataset.

### Problem Scenario:

You are provided with the following details:

- A polynomial divisor is given in binary representation (e.g., 1101).
- The dataset, represented in binary, needs to be transmitted over a noisy communication channel.

### Your task is to:

- Implement a CRC-based error detection system that appends the appropriate CRC checksum to the dataset before transmission.
- Simulate the process of data transmission by potentially introducing errors into the transmitted dataset.
- Implement the receiver's side of the system to verify the integrity of the received dataset using CRC error detection.
- Identify and report any errors that might have occurred during transmission.

# Description ( Problem Statement)

## Input:

- Dataset to be transmitted (binary format).
- Polynomial divisor for CRC (binary format).

## Output:

- Transmitted dataset (potentially with introduced errors).
- Received dataset.
- Detection of whether errors were present or not.
- If errors are detected, identify the positions of errors.

## Example:

Let's consider an example:

## Given:

- Dataset: 11010101101 (binary)
- Polynomial Divisor: 1011 (binary)

## Transmission:

- Sender appends the CRC checksum to the dataset.
- Transmitted dataset: 11010101101100 (binary) [Append the CRC checksum]

## Transmission with Errors:

- Errors are introduced during transmission: 11010101111100 (binary)

## Reception:

- Receiver receives the potentially erroneous dataset: 11010101111100 (binary)
- Receiver performs CRC error detection.
- Errors are detected.

## Error Detection Outcome:

- Errors were detected in the received dataset.
- Error positions: 10 (binary index positions where errors occurred)