PROGRAM 4 : To write a program for error detecting code using CRC-CCITT (16- bits).

```
import java.util.Scanner;

public class CRC_CCITT {
    static int[] input = new int[50];
    static int[] crc   = new int[20];
    static int[] poly  = {1,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,1};  // CRC-16 polynomial
    static int n, r = 16;

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Step 1: Read input message
        System.out.print("Enter the number of bits in the message: ");
        n = sc.nextInt();

        System.out.println("\nEnter the binary message:");
        for (int i = 0; i < n; i++) {
            input[i] = sc.nextInt();
        }

        // Step 2: Append r zeros at the end
        for (int i = n; i < n + r; i++) {
            input[i] = 0;
        }

        // Step 3: Generate CRC
        genCRC();

        // Step 4: Show CRC
        System.out.print("\nGenerated CRC is: ");
        for (int i = 0; i < r; i++) {
            System.out.print(crc[i] + " ");
    input[n + i] = crc[i];   // append CRC bits to message
}

// Step 5: Display transmitted codeword
System.out.print("\nTransmitted code word is: ");
for (int i = 0; i < n + r; i++) {
    System.out.print(input[i] + " ");
}

// Step 6: Ask for error simulation
System.out.print("\n\nDo you want to make an error (1/0): ");
int choice = sc.nextInt();

if (choice == 1) {
    System.out.print("Enter the position: ");
    int pos = sc.nextInt() ;
    input[pos] ^= 1;            // flip bit
}

// Step 7: Display received codeword
System.out.print("\nReceived code word is: ");
for (int i = 0; i < n + r; i++) {
    System.out.print(input[i] + " ");
}

// Step 8: Recalculate CRC
genCRC();

System.out.print("\nGenerated CRC is: ");
for (int i = 0; i < r; i++) {
    System.out.print(crc[i]);
}

// Step 9: Check CRC remainder to detect error
boolean error = false;
```

```
        for (int i = 0; i < r; i++) {
            if (crc[i] != 0) {
                error = true;
                break;
            }
        }

        if (!error)
            System.out.println("\nMessage has no error");
        else
            System.out.println("\nMessage received has an Error");
    }

    // CRC generation using polynomial division
    static void genCRC() {
        int[] temp = new int[50];
        System.arraycopy(input, 0, temp, 0, n + r);

        for (int i = 0; i < n; i++) {
            if (temp[i] == 1) {
                for (int j = 0; j <= r; j++) {
                    temp[i + j] ^= poly[j];
                }
            }
        }
        System.arraycopy(temp, n, crc, 0, r);  // remainder is CRC
    }
}
```

Commands

Create:gedit filename.java

Compile: javac filename.java

Run:java filename

Output:

Case 1: With Error

```
Enter the number of bits in the message: 3

Enter the binary message:
1
1
1

Generated CRC is: 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1
Transmitted code word is: 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1

Do you want to make an error (1/0): 1
Enter the position: 1

Received code word is: 1 1 0 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1
Generated CRC is: 0001000000100001
Message received has an Error
```

Case 2: Without Error

```
Enter the number of bits in the message: 3

Enter the binary message:
1
1
1

Generated CRC is: 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1
Transmitted code word is: 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1

Do you want to make an error (1/0): 0

Received code word is: 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 0 1 1 1
Generated CRC is: 0000000000000000
Message has no error
```