

Cycle 2

13 Write a program for error detecting code using CRC-CCITT (16-bits)

```
* def xor(a,b):
    result = []
    for i in range(1, len(b)):
        if a[i] == b[i]:
            result.append('0')
        else:
            result.append('1')
    return ' '.join(result)

def mod2div(dividend, divisor):
    pick = len(divisor)
    tmp = dividend[0:pick]
    while pick < len(dividend):
        if tmp[0] == '1':
            tmp = xor(divisor, tmp) + dividend[pick]
        else:
            tmp = xor('0' * pick, tmp) + dividend[pick]
        pick += 1
    if tmp[0] == '1':
        tmp = xor(divisor, tmp)
    else:
        tmp = xor('0' * pick, tmp)
    checkword = tmp
    return checkword

def encode(data, key):
    l-key = len(key)
    appended-data = data + '0' * (l-key - 1)
    remainder = mod2div(appended-data, key)
```

```

codeword = data + remainder
print("Remainder :", remainder)
print("Encoded Data[Data + remainder]
      codeword)

return codeword

def decode-data( encoded-data, key)
    remainder = mod2 div( encoded-data,
                           key)
    print("Remainder after decoding:",
          remainder)

    if '1' not in remainder:
        print("No error detected in
              received data")
    else:
        print("Error detected in
              received data")

data = "1001001000100100"
key = "1101"
encoded-data = encode( data, key)
decoded-data = decode-data( encoded-data, key)

```

O/p:

Remainder = 11

encoded-data (data + remainder) =

100100100010010011

Remainder after decoding = 000

No error detected in received data.