---------------------------------------------------------------------------

**Name : Atharva Paliwal**

**Roll No : 40 [5B]**

---------------------------------------------------------------------------

**\*\*\* EXPERIMENT NO: 03 \*\*\***

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**AIM-** Write a program to implement CRC algorithm . Ask the user to input variable length data and generator polynomial. Find the CRC code on the basis of input data. Invert any bit of CRC frame and detect the error at receiver side.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**CODE-**

# Function for Division

def divide(crc\_code,gen\_poly,l):

k=len(gen\_poly) #length of generator poly

i,j=0,k #i=index for div ,j=last index of crc code

while j<=l:

rem='' #initialising rem

div=crc\_code[i:j] #new div value

#if first value of new div is 0 then skip

if div[0]=='0':

i,j=i+1,j+1

continue

for m in range(k):

#XOR 0^0=0 0^1=1

rem=rem+['1','0'][div[m]==gen\_poly[m]]

#updating crc code

crc\_code=crc\_code[:i]+rem+crc\_code[j:]

i,j=i+1,j+1

return crc\_code[l-k+1:] #returning rem

# Driver Code

n=int(input('Enter the length of data : '))

data=input('Enter the Data : ')

gen\_poly=input('Enter Generator Polynomial : ')

#CRC

crc\_code=data+'0'\*(len(gen\_poly)-1)

print('Rough CRC code : ',crc\_code)

rem=divide(crc\_code,gen\_poly,n+len(gen\_poly)-1)

print("Remainder for CRC Code : ", rem)

crc\_code=data+rem

print('CRC code : ',crc\_code)

#reciever side

rec\_data=input('Enter Data at Receiver End : ')

rem=divide(rec\_data,gen\_poly,n+len(gen\_poly)-1)

print("Remainder from Reciever's data :" ,rem)

if rem=='0'\*(len(gen\_poly)-1):

print('No Error.')

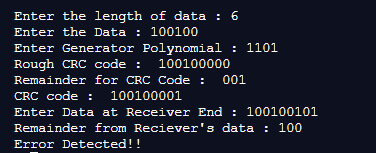
else:

print('Error Detected!!')

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**OUTPUT-**



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*