
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
```

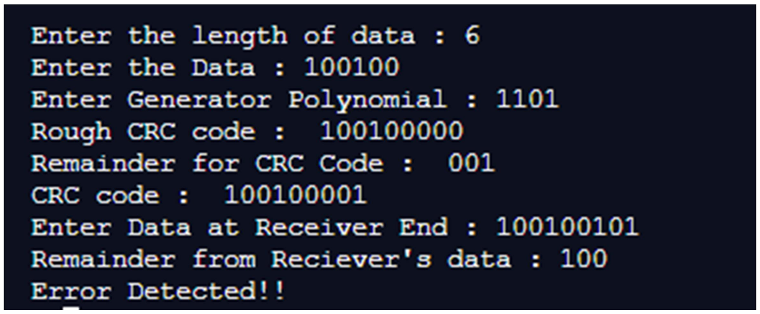
```
# Driver Code
n=int(input('Enter the length of data : '))
data=input('Enter the Data : ')
gen_poly=input('Enter Generator Polynomial : ')
#CRC
crc_code=data+'0'*(len(gen_poly)-1)
print('Rough CRC code : ',crc_code)
rem=divide(crc_code,gen_poly,n+len(gen_poly)-1)
print("Remainder for CRC Code : ", 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-



```
Enter the length of data : 6
Enter the Data : 100100
Enter Generator Polynomial : 1101
Rough CRC code :  100100000
Remainder for CRC Code :  001
CRC code :  100100001
Enter Data at Receiver End : 100100101
Remainder from Reciever's data : 100
Error Detected!!
```

****