17/8/23

CYCLE -2

write a Program for error detecting code using CRC
CCITT(16-bits) $x^{16} + x^{12} + x^5 + x^0$ → GP

Degree

```
#include<stdio.h>
#include<string.h>
#define N strlen(divisor)
char data[30];
char rem[30];
char divisor[10];
int dlength,i,j;
void xor()
{   for(j=1; j<N; j++)
      rem[j] = ((rem[j]==divisor[j])?'0':'1');
}
void crc()
{       for(i=0; i<N; i++)
            rem[i]=data[i];
        do{
            if(rem[0]=='1') xOR();
           for(j=0; j<N-1; j++) rem[j]=rem[j+1];
           rem[j]=data[i++];
        }
       while(i<=dlength+16);
}
void reciver()
{   printf("Enter the data being recived: ");
    scanf("%s", data);
    printf("Data recived: %s", data);
    crc();
    for(i=0; (i<N-1)&&(rem[i]!='1'); i++);
      {if(i<N-1) printf("\n Error detected in data\n");
       else printf("\n No error detected in data\n");
      }
}
```

```
int main()
{ int i=0;
  printf("\n Enter data to be transmitted: ");
  scanf("%s", data);
  printf("\n Enter the Divisor:");
  scanf("%s", divisor);
  dlength = strlen(data);
  for(i = dlength; i < dlength+16; i++)
  {data[i] = '0';}
  printf("\n Data padded with 16 zeros: %s", data);
  crc();
  printf("\n The remainder or CRC is %s", rem);
  for(i = dlength; i < dlength+16; i++)
      data[i] = rem[i-dlength];
  printf("\n Final data being sent: %s", data);
  receiver();
  return 0;
}
```

OUTPUT:

1) Enter data to be transmitted: 1001101
Enter the Divisor: 1011
Data padded with 16 zeroes: 1001101 0000000000000000
The remainder or CRC is: 111
Final data being sent: 1001101 000000000000111
Enter the data being recived: 1001101 000000000000111
Data recived: 1001101 000000000000111
Noo Error detected in data

2) Enter data to be transmitted: 1001101
Enter the Divisor: 1011
Data padded with 16 zeroes: 1001101 0000000000
The remainder or CRC is: 111
Final data being sent: 1001101 000000000000111
Enter the data being recived: 1001101 000000000000111
Data recived: 1001101 000000000000111
Error detected in data

```c
#include<stdio.h>
#include<string.h>
#define N strlen(divisor)
char data[28];
char rem[28];
char divisor[10];
int dlength,i,j;
void XOR(){
    for(j = 1;j < N; j++)
    rem[j] = (( rem[j] == divisor[j])?'0':'1');

}

void receiver(){

    printf("Enter the received data: ");
    scanf("%s", data);
    printf("\n\n");
    printf("Data received: %s", data);

    crc();

    for(i=0;(i<N-1) && (rem[i]!='1');i++);
        if(i<N-1)
            printf("\nError detected\n\n");
        else
            printf("\nNo error detected\n\n");
}

void crc(){

    for(i=0;i<N;i++)
        rem[i]=data[i];
    do{

        if(rem[0]=='1')
            XOR();

        for(j=0;j<N-1;j++)
            rem[j]=rem[j+1];

        rem[j]=data[i++];
    }
    while(i<=dlength+16);

}

int main()
{  int c=0;

    printf("\nEnter data to be transmitted: ");
    scanf("%s",data);
    printf("\n Enter the Divisor: ");
    scanf("%s",divisor);
    dlength=strlen(data);
    for(i=dlength;i<dlength+16;i++)
        data[i]='0';
    printf("\n");
    printf("\n Data padded with 16 zeros : %s",data);
    printf("\n");
    crc();
    printf("\nCRC or Check value is : %s",rem);
    printf("\n rem strlen is : %d ", strlen(rem));
    for(i=dlength+13;i<dlength+16;i++)
       {    //printf("\n %s",data);
           data[i]= rem[c++];
           }
    printf("\n");

    printf("\n Final data to be sent : %s",data);
    printf("\n\n");

    receiver();
        return 0;
}
```

```
Enter data to be transmitted: 1001101

Enter the Divisor: 1011


Data padded with 16 zeros : 10011010000000000000000

CRC or Check value is : 111

Final data to be sent : 10011010000000000000111

Enter the received data: 10011010000000000000111


Data received: 10011010000000000000111
No error detected
```

```
Enter data to be transmitted: 1001101

Enter the Divisor: 1011


Data padded with 16 zeros : 10011010000000000000000

CRC or Check value is : 111

Final data to be sent : 10011010000000000000111

Enter the received data: 10011000000000000000111


Data received: 10011000000000000000111
Error detected
```