Program: Cyclic Redundancy Check(CRC) Code For Error Detection

Code:

```
#include<stdio.h>
#include<string.h>
#define N strlen(gen_poly)
char data[30],check_value[30], gen_poly[10];
int data_length,i,j;

void XOR(){
   for(j = 1;j < N; j++)
   check_value[j] = (( check_value[j] == gen_poly[j])?'0':'1');
}

void crc(){
   for(i=0;i<N;i++)
      check_value[i]=data[i];
   do{

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

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

      check_value[j]=data[i++];
   }while(i<=data_length+N-1);

}

void receiver(){
```

```
    printf("\n \nEnter the received data: ");
    scanf("%s", data);

    crc();

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


int main()
{
    printf("\n Niyati's program for CRC Code");
    printf("\nEnter data to be transmitted: ");
    scanf("%s",data);
    printf("\n Enter the Generating polynomial aka divisor: ");

    scanf("%s",gen_poly);

    data_length=strlen(data);

    for(i=data_length;i<data_length+N-1;i++)
        data[i]='0';

    printf("\n Data padded with n-1 zeros aka divident : %s",data);
```

```
    crc();

    printf("\nCRC or Check value is : %s",check_value);

    for(i=data_length;i<data_length+N-1;i++)
        data[i]=check_value[i-data_length];


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

    receiver();
        return 0;
}
```

Output:

Niyati's program for CRC Code

Enter data to be transmitted: 101001001

Enter the Generating polynomial aka divisor: 101011

Data padded with n-1 zeros aka divident : 10100100100000

CRC or Check value is : 10111

Final data to be sent : 10100100110111

Enter the received data: 11100100110111

Error detected