Suman Ku Paul
1BM18CS111

```
#include <stdio.h>
 char m[50], g[50], r[50], q[50], temp[50];
     Void calframe (int);
      Void crc (int);
    Void calram ();
    Void shiftl();
    int main()
{     int n, i=0,
     Char ch, flag =0;
      Printf (" Enter fram bits:");
    While ((ch=getc(stdin))!='\n')
      m [i++]=ch;
      n=i
    for (i=0; i<16; i++)
      m [n++] = '0';
    m[n] = '\0';
   Print f (" Message after appending 16 zeros: %s",m);
    for (i=0; i<=16; i++)
       g[i]='0';
 g[0]=g[5]=g[11]= g[16]='1'; g[17]='0';
   printf (" \ngeneration : %s\n",g);

   crc(n);
    Printf ("\n\nquotient :%s",q);
     Cal frame (n);
 Print f ("\ntransmitted fram : %s", m);
   print f ( "\nEnter transmitted frem :");
      Scanf (" \n%s", m);
       Print (" CRC checking\n");
     Crc (n);
Printf ("\nlast remainder : %s"),r);
    for (i=0, i<16; i++)
      if (r [i] != '0')
        flag = 1;
      else
      Continue ;
      if (flag ==1)
  Printf ("Error during transmission");
    else

  Pf (" \n) Received frame is Correct")}
   } Void crc (int n)
    { int i,j;
  for (i=0, i<n, i++)
      temp [i] = m [i];
    for (i=0; i<16; i++)
      r [i] = m [i];
   Printf ("\nintermediate remainder\n");
    for (i=0; i<n-16; i++)
      { if r [0] == '1')
        { q [i] = '1';
      Calram ();
      } else { q [i] = '0';
              shift (); }
        r [16] = m [17+i];
        r [17] = '\0';
   Printf ("\n remainder %d: %s", i+1, r);
```

```
for (j=0; j<17; j++)
    temp[j] = r[j];
  }
      g[n-16] = '\0';
    }
  Vaid  Calran()
    {     int i,j;
  for (i=1; i<=16; i++)
r[i-1] = ((int)temp[i]-48)^((int)g[i]-48)+48;
  }
  Vaid  shift()
    {  int i;
       for (i=1; i<=16; i++)
       r[i-1] = r[i];
    }
     Vaid  Caldran (int n)
     {
          int i, k=0;
        for (i=n-16; i<n; i++)
    m[i] = ((int)m[i]-48)^(((int)r[k++])-48)
                                              +48;
      m[i] = '\0';

      }
```