

Write a program for error detecting code using CRC²CCITT (16-bits).

```

write a program for error detecting code using CRC2CCITT
#include <stdio.h>
char m[50], g[50], r[50], q[50], temp[50];
void caltrans(int);
void crc(int);
void calnam();
void shiftl();
void main()
{
    int n, i = 0;
    char ch, flag = 0;
    printf("Enter the frame bits:");
    while ((ch = getch()) != '\n')
        m[i++] = ch;
    n = i;
    for (i = 0; i < 16; i++)
        m[n++] = '0';
    m[n] = '\0';
    printf("Message after appending 16 zeros : %s", m);
    for (i = 0; i < 16; i++)
        g[i] = '0';
    g[0] = g[4] = g[12] = g[16] = '1';
    g[17] = '\0';
    printf("\n Generator : %s", g);
    crc(n);
    printf("\n Quotient : %s", q);
    caltrans(n);
    printf("\n Transmitted frame : %s", m);
    printf("\n Enter received frame:");
    scanf("%s", r);
    printf("\n CRC checking\n");
    crc(n);
    printf("\n Last remainder: %s", r);
    for (i = 0; i < 16; i++)
        if (r[i] != '0')
            flag = 1;
    if (flag)
        printf("Error detected\n");
    else
        printf("No error detected\n");
}

```

```

if (flag == 1)
    printf("Error during transmission");
else
    printf("In Received frame is correct");
}

```

```

void enc (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];
    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] = '10';
        for (j = 0; j <= 17; j++)
            temp[j] = r[j];

```

```

    }
    q[n-16] = '10';
}

```

```

void calram()
{

```

```

    int i, j;
    for (i = 1; i <= 16; i++)
        r[i-1] = ((int)temp[i]-48) ^ ((int)q[i-48]+48);
}

```

```

void shiftLeft()
{
    int i;
    for (i=1; i<=16; i++)
        s[i-1] = s[i];
}

void caltrans(int n)
{
    int i, k=0;
    for (i=n-16; i<n; i++)
        m[i] = ((int)m[i]-48)^(int)s[k++]-48)+48;
    m[i] = '\0';
}

```

Output:

Enter the frame bits: 10110110110010110101011
 message after appending 16 zeros: 10110000000000000000
 generator: 1000100000100001
 quotient: 1011
 Transmitted frame: 10110110110010110101011
 Enter transmitted frame: 10110110110010110101011
 CRC checking
 last remainder: 0000000000000000
 Received frame is correct.

Output:

```
Enter the frame bits:1011
Message after appending 16 zeros:10110000000000000000
generator:10001000000100001

quotient:1011
transmitted frame:10111011000101101011
Enter transmitted freme:10111011000101101011
CRC checking

last remainder:0000000000000000

Received freme is correct|
```