WEEK 13

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

CODE:

```
#include<stdio.h>
int arr[17];

void xor(int x[], int y[])
{
    int k=0;
    for(int i=1;i<16;i++)
    {
        if(x[i]==y[i])
            arr[k++]=0;
        else
            arr[i]=1;
    }
}

void main()
{
    int dd[17],div[33],ze[17],i,k;

    printf("Enter the dataword \n");
    for(i=0;i<17;i++)
        scanf("%d",&div[i]);

    for(i=i;i<33;i++)
        div[i]=0;

    for(i=0;i<17;i++)
        ze[i]=0;
    printf("Enter dividend \n");
```

```
for(i=0;i<17;i++)
    scanf("%d",&dd[i]);

i=0;
k=0;
    for(i=i;i<17;i++)
        arr[k++]=div[i];
while(i<33)
{
    if(arr[0]==0)
        xor(arr,ze);
    else
        xor(arr,dd);

    arr[16]=div[i++];

}
k=0;
for(i=17;i<33;i++)
    div[i]=arr[k++];
printf("Codeword: ");
    for(i=0;i<33;i++)
        printf("%d",div[i]);

for(i=0;i<17;i++)
    arr[i]=0;

printf("\nAt receiver end \n");

 k=0;
    for(i=i;i<17;i++)
        arr[k++]=div[i];
while(i<33)
{
    if(arr[0]==0)
```

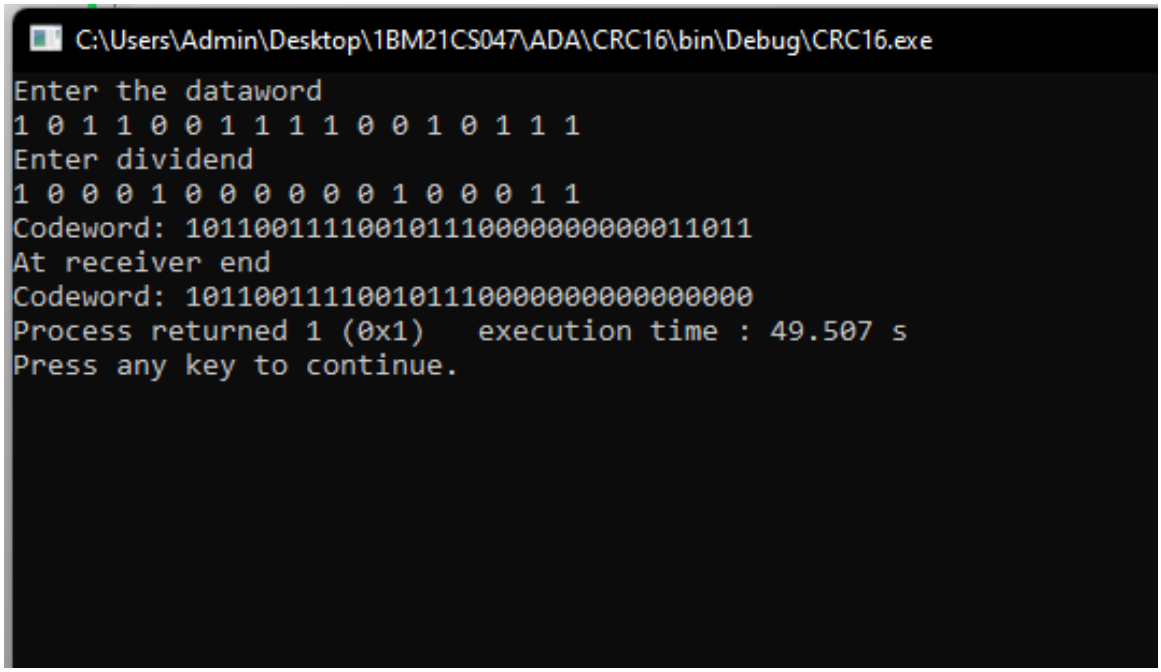
```
            xor(arr,ze);
        else
            xor(arr,dd);

        arr[16]=div[i++];

    }
    k=0;
    for(i=17;i<33;i++)
        div[i]=arr[k++];

    printf("Codeword: ");
        for(i=0;i<33;i++)
            printf("%d",div[i]);
}
```

OUTPUT:



OBSERVATION:

Experiment -13

Date 17/8/23
Page

Aim: To write a program for error detection using CRC-CCITT (16-bits)

```
#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+N-1);
}

void receiver(){
    printf("Enter data to be received:");
    scanf("%d", &data);
```

```
    printf("Data received : %s", data);
    recc();
    for(i=0; i<N-1 && rem[i] != '1'; i++);
        if (i<N-1)
            printf("Error in transmission");
        else
            printf("No error in transmission");
}

int main()
{
    printf("Enter data to be transmitted");
    scanf("%s", &data);
    printf("Enter divisor");
    scanf("%s", &divisor);
    dlength = strlen(data);
    for(i=dlength; i<dlength+N-1; i++)
        data[i] = '0';
    recc();
    for(i=dlength; i<dlength+N-1; i++)
        data[i] = rem[i-dlength];
    printf("Data being sent : %s", data);
    receiver();
    return 0;
}
```

Output :

Enter data to be transmitted : 1100101011001001

Enter divisor : 10001000000100001

Data being sent : 11101001011100001

Data received : 11101001011100001

No error in transmission.