SONA COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

B.E / B. Tech DEGREE SEMESTER END PRACTICAL EXAMINATIONS, DEC 2020 – JAN 2021

REGISTER NUMBER	:	**1**	**5**	**1**	**8**	**1**	**0**	**2**	**1**	**2**	**6**
SUBJECT CODE	:	U15CS505R									
SUBJECT NAME	:	COMPUTER NETWORKS LABORATORY									
DATE	:	0	6	/	0	1	/	2	0	2	1
BATCH	:	2									

TYPE OR COPY / PASTE YOUR ALLOTTED QUESTION DOWN HERE IN THE BOX GIVEN BELOW

Machine ‘A’ sends “1101011101” to machine ‘B’. To add the error detection and correction code machine ‘A’ uses “10110” as generator polynomial. Write a java program to add the CRC code with the message at machine ‘A’ and verify the appended CRC at machine ‘B’

Aim:

To write a java program to add the CRC code with the message at 'A' machine and to verify the added CRC code at machine 'B'.

Algorithm:

Step 1: Start the program.

Step 2: get the data size and data to be sent from A as input.

Step 3: Accept the divisor data bit by bit.

Step 4: Divide the data to be sent by the divisor inputted.

Step 5: Remainder array stores the remainder, initially the bits will be set to data bits.

Step 6: Run a loop for the number of bits of data and exor the bits of remainder and divisor.

Step 7: The last bit of remainder is taken from data which is the "carry" taken from the dividend after every step of division

Step 8: In receive method get the data (Generate CRC) and divisor

Step 9: Check whether the remainder is 0 or not.
If the remainder is 0 then there is no error in data
else there is an error in received data.

Step 10: Display the output and stop the program.

Program:

```
import java.util.*;
class crc {
    Public Static void main (String args[])
    {
     Scanner sc = new Scanner(System.in);
     System.out.print("Enter size of data : ");
     int n = sc.nextInt();
     int data[] = new int[n];
     System.out.println("Enter the data bit by bit :");
     for(int i=0 ; i<n; i++)
     {
       System.out.println("Enter bit number"+(n-i)+":");
        data[i] = sc.nextInt();
     }
    System.out.println("Enter the size of divisor :");
    n = sc.nextInt();
    int divisor[] = new int[n];
    System.out.println("Enter the divisor bit by bit ");
    for(int i=0; i<n; i++)
    {
     System.out.println("Enter bit number "+(n-i)+":");
       divisor[i] = sc.nextInt();
    }
```

```
int remainder[] = divide(data, divisor);
for(int i=0; i< remainder.length-1; i++)
{
    System.out.println(remainder[i]);
}
System.out.println("\n The CRC code generated is : ");
for(int i=0; i< data.length; i++)
{
   System.out.print(data[i]);
}
for (int i=0; i< remainder.length -1; i++)
{
    System.out.print(remainder[i]);
}
int Sent_data[] = new int [data.length + remainder.length -1];
System.out.println("\n Enter the data to be sent : ");
for(int i=0; i< Sent_data.length; i++)
 {
   System.out.println("Enter bit number " + (Sent_data.length -i) + " : ");
      Sent_data[i] = sc.nextInt();
 }
receive(sent_data, divisor);
}
```

```
Static int [] divide (int old-data[], int divisor [])
{
    int  remainder [], data[];
    data = new  int [old-data.length + divisor.length];
    System. arraycopy (old-data , 0, data , 0, old-data.length );

    remainder  = new  int [divisor. length ];
   System. arraycopy ( data , 0, remainder , 0, divisor. length );

   for(int i=0 ; i< old-data. length ; i++)
    {
     System. out. println ((i+1) + ". First data bit is : " remainder[0]);
     System. out. println (" Remainder : ");
     if (remainder[0] == 1)
      {
        for (int j=1 ; j< divisor. length ; j++)
        {
          remainder [j-1] = exor(remainder[j], divisor[j]);
          System. out. print(remainder [j-1]);
        }
      }
     else {
        for (int j=1 ; j< divisor. length ; j++)
        {
          remainder[j-1] = exor (remainder[j],0);
          System. out. print ( remainder [j-1]);
        }
   }
```

```
        remainder[divisor.length-1] = data[i+divisor.length];
        System.out.println(remainder[divisor.length-1]);
    }
    return remainder;
}
Static int exor(int a, int b) {
        if (a==b)
        {
            return 0;
        }
        else {
            return 1;
        }
}
Static void recieve (int data[], int divisor[]) {
        int remainder[] = divide(data, divisor);
        for (int i=0; i< reminder.length; i++)
        {
            if (remainder[i]!=0)
            {
                System.out.println("There is an error in recieved data");
                return;
            }
        }
        System.out.println("Data was received without any error");
    }
}
```

Output:

```
> Javac crc.java
> Java crc
Enter the Size of data :
10
Enter data bit by bit :
Enter bit number 10 :
1
Enter bit number 9:
1
Enter bit number 8:
0
Enter bit number 7:
1
Enter bit number 6:
0
Enter bit number 5:
1
Enter bit number 4:
1
Enter bit number 3:
1
Enter bit number 2:
0
Enter bit number 1:
1
Enter the size of divisor:
5
```

Enter bit number 5 :
1
Enter bit number 4:
0
Enter bit number 3:
1
Enter bit number 2:
1
Enter bit number 1:
0

1. The First data bit is : 1
   Remainder : 11001
2. First data bit is : 1
   Remainder : 11111
3. First data bit is : 1
   Remainder : 10011
4. First data bit is : 1
   Remainder : 01010
5. First data bit is 0
   Remainder : 10101
6. First data bit is 1
   Remainder : 00110
7. First data bit is 0
   Remainder : 01100
8. First data bit is 0
   Remainder : 11000
9. First data bit is 1
   Remainder : 11100
10. First data bit is 1
    Remainder : 10100

1010

The CRC code generated is

11010111011010.

Enter data to be sent:

Enter bit number 14: 1

Enter bit number 13: 1

Enter bit number 12: 0

Enter bit number 11: 1

Enter bit number 10: 0

Enter bit number 9: 1

Enter bit number 8: 1

Enter bit number 7: 1

Enter bit number 6: 0

Enter bit number 5: 1

Enter bit number 4: 1

Enter bit number 3: 0

Enter bit number 2: 1

Enter bit number 1: 0

1) First data bit is : 1
   Remainder : 11001

2) First data bit is : 1
   Remainder : 11111

3) First data bit is : 1
   Remainder : 10011

4) First data bit is 1
   Remainder : 01010

5) First data bit is 0
Remainder : 10101

6) First data bit is : 1
Remainder : 00111

7) First data bit is : 0
Remainder : 01110

8) First data bit is 0
Remainder : 11101

9) First data bit is 1
Remainder 10110

10) First data bit is 1
Remainder 00000

11) First data bit is 0
Remainder 00000

12) First data bit is 0
Remainder : 00000

13) First data bit is 0
Remainder : 00000

14) First data bit is 0
Remainder : 00000

Data was received without any error.

Result:

The above program code is implemented and the data sent to B from A is verified to errorless.