

SHIVANI GUPTA

2018UIT2586

**Ques 1: Implement various error detection techniques like parity check , check sum and CRC .**

**CHECKSUM CODE :**

```
#include <iostream>
#include <string.h>

using namespace std;

int main()
{
    char a[20], b[20];
    char sum[20], complement[20];
    int i;

    cout << "Enter first binary string\n";
    cin >> a;
    cout << "Enter second binary string\n";
    cin >> b;

    if (strlen(a) == strlen(b))
    {
        char carry = '0';
        int length = strlen(a);

        for (i = length - 1; i >= 0; i--)
        {
            if (a[i] == '0' && b[i] == '0' && carry == '0')
            {
                sum[i] = '0';
                carry = '0';
            }
            else if (a[i] == '0' && b[i] == '0' && carry == '1')
            {
                sum[i] = '1';
                carry = '0';
            }
            else if (a[i] == '0' && b[i] == '1' && carry == '0')
            {
                sum[i] = '1';
            }
        }
    }
}
```

```

        carry = '0';
    }
    else if (a[i] == '0' && b[i] == '1' && carry == '1')
    {
        sum[i] = '0';
        carry = '1';
    }
    else if (a[i] == '1' && b[i] == '0' && carry == '0')
    {
        sum[i] = '1';
        carry = '0';
    }
    else if (a[i] == '1' && b[i] == '0' && carry == '1')
    {
        sum[i] = '0';
        carry = '1';
    }
    else if (a[i] == '1' && b[i] == '1' && carry == '0')
    {
        sum[i] = '0';
        carry = '1';
    }
    else if (a[i] == '1' && b[i] == '1' && carry == '1')
    {
        sum[i] = '1';
        carry = '1';
    }
    else
        break;
}
cout << "\nSum=" << carry << sum;

for (i = 0; i < length; i++)
{
    if (sum[i] == '0')
        complement[i] = '1';
    else
        complement[i] = '0';
}

if (carry == '1')
    carry = '0';
else
    carry = '1';

cout << "\nChecksum = " << carry << complement;
}
else

```

```
        cout << "\nWrong input strings";  
  
    return 0;  
}
```

## OUTPUT :

```
Enter first binary string  
101010  
Enter second binary string  
111000  
  
Sum=1100010  
Checksum = 0011101  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

## CYCLIC REDUNDANCY CHECK CODE :

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int da[20], di[20], te[20], tem[20], l;
    int i, j, m, n, data, div, t, k, e;
    printf("\nEnter the total bit of data and divisor");
    scanf("%d %d", &data, &div);
    m = data + div - 1;
    printf("\nEnter the data:");
    for (i = 0; i < data; i++)
    {
        scanf("%d", &da[i]);
        te[i] = da[i];
    }
    for (i = data; i < m; i++)
    {
        te[i] = 0;
    }
    printf("\nEnter the divisor");
    for (i = 0; i < div; i++)
    {
        scanf("%d", &di[i]);
    }
    l = div;
    t = 0;
    k = 0;
    for (i = 0; i < data; i++)
    {
        e = 0;
        t = 0;
        for (j = 1; j < div; j++)
        {
            if (((da[j] == 1) && (di[j] == 1)) || ((da[j] == 0) && (di[j] == 0
)))
            {
                tem[j - 1] = 0;
                if (e != 1)
                {
                    k = k + 1;
                    t = t + 1;
                    i = i + 1;
                }
            }
            else
            {

```

```

        tem[j - 1] = 1;
        e = 1;
    }
}
j = 0;
for (e = t; e < div - 1; e++)
{
    da[j] = tem[e];
    j++;
}
for (j = j; j < div; j++)
{
    if (l >= data + 1)
    {
        da[j] = 0;
    }
    else
    {
        da[j] = te[l];
        l = l + 1;
    }
}
}
printf("\n The CRC BITS are\t ");
for (i = 0; i < div - 1; i++)
{
    printf(" %d", tem[i]);
}
}

```

**OUTPUT :**

```

Enter the total bit of data and divisor
6 4

Enter the data: 1 0 0 1 0 0

Enter the divisor 1 1 0 1

The CRC BITS are      0 0 1

...Program finished with exit code 0
Press ENTER to exit console.

```

## PARITY CHECK CODE :

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n, c = 0, m = 0;
    cout << "Enter the no. of bits in sending message : ";
    cin >> n;
    char send[n + 1], receive[n + 1];
    for (int i = 0; i < n; i++)
    {
        cin >> send[i];
        if (send[i] == '1')
            c++;
    }
    if (c % 2 == 0)
        send[n] = '0';
    else
        send[n] = '1';
    cout << "\nNew sender message after adding parity : ";
    for (int i = 0; i < n + 1; i++)
        cout << send[i];
    cout << "\nEnter the received message : ";
    for (int i = 0; i < n + 1; i++)
    {
        cin >> receive[i];
        if (receive[i] == '1')
            m++;
    }
    if (m % 2 == 0)
        cout << "\nThe received message has even no. of 1's ";
    else
        cout << "\nThere is some error in the message ";
}
```

## OUTPUT :

```
Enter the no. of bits in sending message : 7
1 0 0 1 0 0 1

New sender message after adding parity : 10010011
Enter the received message : 10010111

There is some error in the message

...Program finished with exit code 0
Press ENTER to exit console.█
```

```
Enter the no. of bits in sending message : 7
1 1 1 1 1 1 1

New sender message after adding parity : 11111111
Enter the received message : 11111111

The received message has even no. of 1's

...Program finished with exit code 0
Press ENTER to exit console.█
```

**Ques 2 : Implement error correcting technique like hamming code .**

### **HAMMING CODE :**

```
#include <iostream>
using namespace std;
int main()
{
    int data[10];
    int dataatrec[10], c, c1, c2, c3, i;
    cout << "Enter 4 bits of data one by one\n";
    cin >> data[0];
    cin >> data[1];
    cin >> data[2];
    cin >> data[4];
    data[6] = data[0] ^ data[2] ^ data[4];
    data[5] = data[0] ^ data[1] ^ data[4];
    data[3] = data[0] ^ data[1] ^ data[2];

    cout << "\nEncoded data is\n";
    for (i = 0; i < 7; i++)
        cout << data[i];
    cout << "\n\nEnter received data bits one by one\n";
    for (i = 0; i < 7; i++)
        cin >> dataatrec[i];
    c1 = dataatrec[6] ^ dataatrec[4] ^ dataatrec[2] ^ dataatrec[0];
    c2 = dataatrec[5] ^ dataatrec[4] ^ dataatrec[1] ^ dataatrec[0];
    c3 = dataatrec[3] ^ dataatrec[2] ^ dataatrec[1] ^ dataatrec[0];
    c = c3 * 4 + c2 * 2 + c1;

    if (c == 0)
    {
        cout << "\nNo error while transmission of data\n";
    }
    else
    {
        cout << "\nError on position " << c;
        cout << "\nData sent : ";
        for (i = 0; i < 7; i++)
            cout << data[i];
        cout << "\nData received : ";
        for (i = 0; i < 7; i++)
            cout << dataatrec[i];
        cout << "\nCorrect message is\n";
        if (dataatrec[7 - c] == 0)
            dataatrec[7 - c] = 1;
    }
}
```



```
        else
            dataatrec[7 - c] = 0;
        for (i = 0; i < 7; i++)
        {
            cout << dataatrec[i];
        }
    }
    return 0;
}
```

## OUTPUT :

```
Enter 4 bits of data one by one
0 1 1 1

Encoded data is
0110100

Enter received data bits one by one
0 1 1 0 1 0 0

No error while transmission of data

...Program finished with exit code 0
Press ENTER to exit console.□
```