Experiment No: 03
Write a program for error detection and correction for 7/8 bits ASCII codes using
Hamming Codes or CRC. Demonstrate the packets captured traces using Wireshark
Packet Analyzer Tool for peer-to-peer mode.

Program for Hamming Code in C++:

```cpp
#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[7];
      cin>>data[6];
      cin>>data[5];
      cin>>data[3];

      //Calculation of even parity
      data[4]=data[5]^data[6]^data[7];
      data[2]=data[3]^data[6]^data[7];
      data[1]=data[3]^data[5]^data[7];

      cout<<"\nEncoded data is\n";
      for(i=1;i<=7;i++)
         cout<<data[i];

      cout<<"\n\nEnter received data bits one by one\n";
      for(i=1;i<=7;i++)
         cin>>dataatrec[i];

      c1=dataatrec[1]^dataatrec[3]^dataatrec[5]^dataatrec[7];
      c2=dataatrec[2]^dataatrec[3]^dataatrec[6]^dataatrec[7];
      c3=dataatrec[4]^dataatrec[5]^dataatrec[6]^dataatrec[7];
      c=c3*4+c2*2+c1;
      if(c==0)
      {
      cout<<"\ncongratulations there is no error: ";
   }
   else
   {
      cout<<"\nerror on the postion:"<<c;
      cout<<"\nCorrect message is:";
      if(dataatrec[c]==0)
            dataatrec[c]=1;
      else
```

```
        dataatrec[c]=0;
     for (i=1;i<=7;i++)
     {
        cout<<dataatrec[i];
     }
  }

     return 0;
}
```

Output:

Program for CRC in C++:

```cpp
#include <iostream>
using namespace std;

int main()
{
   int i,j,k,l;

     //Get Frame
     int fs;
     cout<<"\n Enter Frame size: ";
     cin>>fs;
```

```cpp
int f[20];

cout<<"\n Enter Frame:";
for(i=0;i<fs;i++)
{
cin>>f[i];
}

//Get Generator
int gs;
cout<<"\n Enter Generator size: ";
cin>>gs;

int g[20];

cout<<"\n Enter Generator:";
for(i=0;i<gs;i++)
{
cin>>g[i];
}
cout<<"\n Sender Side:";
cout<<"\n Frame: ";
for(i=0;i<fs;i++)
{
cout<<f[i];
}
cout<<"\n Generator :";
for(i=0;i<gs;i++)
{
cout<<g[i];
}
//Append 0's
int rs=gs-1;
```

```cpp
cout<<"\n Number of 0's to be appended: "<<rs;
for (i=fs;i<fs+rs;i++)
{
f[i]=0;
}

int temp[20];
for(i=0;i<20;i++)
{
temp[i]=f[i];
}
cout<<"\n Message after appending 0's :";
for(i=0; i<fs+rs;i++)
{
cout<<temp[i];
}

//Division
for(i=0;i<fs;i++)
{
j=0;
k=i;
//check whether it is divisible or not
if (temp[k]>=g[j])
{
   for(j=0,k=i;j<gs;j++,k++)
   {
      if((temp[k]==1 && g[j]==1) || (temp[k]==0 && g[j]==0))
      {
         temp[k]=0;
      }
      else
      {
         temp[k]=1;
```

```cpp
        }
    }
}
}

//CRC
int crc[15];
for(i=0,j=fs;i<rs;i++,j++)
{
crc[i]=temp[j];
}
cout<<"\n CRC bits: ";
for(i=0;i<rs;i++)
{
cout<<crc[i];
}

cout<<"\n Transmitted Frame: ";
int tf[15];
for(i=0;i<fs;i++)
{
tf[i]=f[i];
}
for(i=fs,j=0;i<fs+rs;i++,j++)
{
tf[i]=crc[j];
}
for(i=0;i<fs+rs;i++)
{
cout<<tf[i];
}

cout<<"\n Receiver side : ";
cout<<"\n Received Frame: ";
```

```cpp
for(i=0;i<fs+rs;i++)
{
cout<<tf[i];
}


for(i=0;i<fs+rs;i++)
{
temp[i]=tf[i];
}


//Division
for(i=0;i<fs+rs;i++)
{
j=0;
k=i;
if (temp[k]>=g[j])
{
    for(j=0,k=i;j<gs;j++,k++)
    {
        if((temp[k]==1 && g[j]==1) || (temp[k]==0 && g[j]==0))
        {
            temp[k]=0;
        }
        else
        {
            temp[k]=1;
        }
    }
}
}


cout<<"\n Reaminder: ";
int rrem[15];
for (i=fs,j=0;i<fs+rs;i++,j++)
```

```cpp
{
rrem[j]= temp[i];
}
for(i=0;i<rs;i++)
{
cout<<rrem[i];
}

int flag=0;
for(i=0;i<rs;i++)
{
if(rrem[i]!=0)
{
    flag=1;
}
}

if(flag==0)
{
cout<<"\n Since Remainder Is 0 Hence Message Transmitted From Sender To Receriver Is Correct";
}
else
{
cout<<"\n Since Remainder Is Not 0 Hence Message Transmitted From Sender To Receriver Contains Error";
}
return 0;
}
```

```
C:\Users\NIS\Documents\crc.exe

 Enter Frame size: 9

 Enter Frame:1
1
0
0
1
0
1
0
1

  Enter Generator size: 5

  Enter Generator:1
0
1
0
1

  Sender Side:
  Frame: 110010101
  Generator :10101
  Number of 0's to be appended: 4
  Message after appending 0's :1100101010000
  CRC bits: 1011
  Transmitted Frame: 110010101011011
  Receiver side :
  Received Frame: 110010101011011
  Reaminder: 0000
  Since Remainder Is 0 Hence Message Transmitted From Sender To Receriver Is Corr
ect
---------------------------------
Process exited after 85.5 seconds with return value 0
Press any key to continue . . .
```