‘Hamming Code’ method to encode data at sender side and decoding the data at receiver end for error correction.

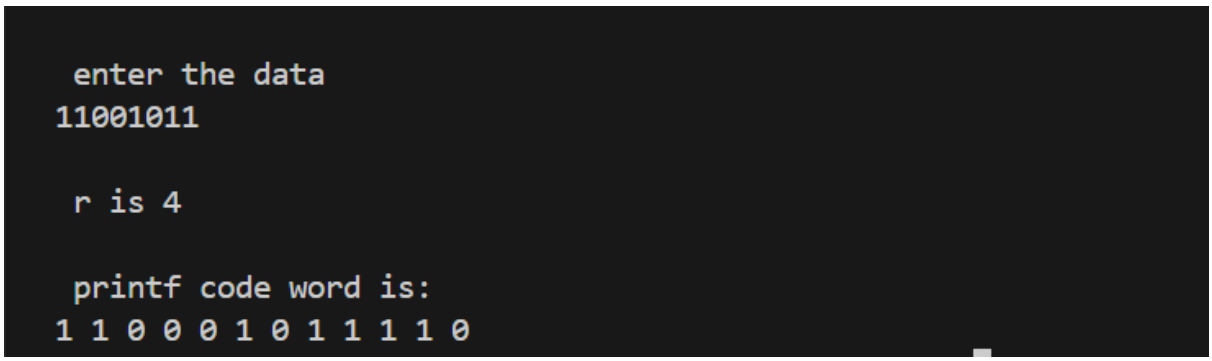
Sender Side :

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

int main() {
   int flag, len, bit;

      // *************START  CODE FOR hamming SENDER ************//
   char data[100];
   int data1[100],data2[100];
   int dl,r,i,j,k,z,c;
   char buf[100];
   memset(data,'\0',100);
   printf("\n enter the data \n");
   scanf("%s",data);

   dl=strlen(data);
   i=0;
   while(1)
   {
      if(pow(2,i)>=i+dl+1)
         break;
      i++;
   }
```

```
r=i;
printf("\n r is %d \n",r);
for(i=0;i<dl;i++)
{
   data1[i]=data[i]-48;
}
for(i=0;i<r;i++)
   {
      z=pow(2,i);
      data2[z]=999;
   }
   j=0;
for(i=dl+r;i>=1;i--)
{
   if(data2[i]!=999){
      data2[i]=data1[j];
      j++;
      }
}

for(i=0;i<r;i++)
{
   z=pow(2,i);
   c=0;
   for(j=z;j<=dl+r;j=z+k)
   {
      for(k=j;k<z+j;k++)
      {
         if(k<=r+dl)
         if(data2[k]!=999)
            c=c+data2[k];
```

```
        }
      }
      data2[z]=c%2;
   }

   printf("\n printf code word is: \n");
   j=0;
   for(i=dl+r;i>=1;i--){
      printf("%d ",data2[i]);
      buf[j]=data2[i]+48;
      j++;
       }
// **************END CODE FOR hamming SENDER ************//

return(0);
}
```

Output :



Receiver Side :

```
#include <stdio.h>
#include <math.h>
#include <string.h>

int main() {

   char data[100];
   int data1[100],data2[100];
   int dl,r,i,j,k,z,c,l;
   printf("\n enter the received data \n");
   scanf("%s",data);

//************** Hamming check code start *************

   dl=strlen(data);
   i=0;
   while(1)
   {
      if(pow(2,i)>=dl+1)
         break;
      i++;
   }
   r=i;
   j=dl-1;
   for(i=1;i<=dl;i++)
   {
      data1[i]=data[j]-48;
      j--;
   }
```

```
l=1;
data2[0]=0;
for(i=0;i<r;i++)
{
   z=pow(2,i);
   c=0;
   for(j=z;j<=dl;j=z+k)
   {
      for(k=j;k<z+j;k++)
      {
         if(k<=dl)
            c=c+data1[k];
      }
   }
   data2[l]=c%2;
   data2[0]=data2[0]+data2[l];
   l++;
}

if(data2[0]==0)
   {
      printf("\n data recv ok \n");

   }
else
   {
      printf("\n data recv wrong \n");
      j=0;
      for(i=r;i>=1;i--)
      {
```

```
            if(data2[i]==1)
            j=j+pow(2,(i-1));
        }
        printf("\n error at position %d",j);
        if(data1[j]==0)
            data1[j]=1;
        else
            data1[j]=0;
        printf("\n corrected codeword is: ");
        for(i=dl;i>=1;i--)
            printf("%d ",data1[i]);
        printf("\n");
    }

//************** Hamming code check complete *************

return(0);
}
```

Output :