Hamming Code

```
#include<stdio.h>

int main() {
   int data[10];
   int dataatrec[10],c,c1,c2,c3,i;

   printf("Enter 4 bits of data one by one\n");
   scanf("%d",&data[0]);

   scanf("%d",&data[1]);
   scanf("%d",&data[2]);
   scanf("%d",&data[4]);

   //Calculation of even parity
   data[6]=data[0]^data[2]^data[4];
data[5]=data[0]^data[1]^data[4];
data[3]=data[0]^data[1]^data[2];

printf("\nEncoded data is\n");
for(i=0;i<7;i++)
      printf("%d",data[i]);

   printf("\n\nEnter received data bits one by one\n");
   for(i=0;i<7;i++)
      scanf("%d",&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) {
printf("\nNo error while transmission of data\n");
   }
else {
printf("\nError on position %d",c);

printf("\nData sent : ");
       for(i=0;i<7;i++)
        printf("%d",data[i]);

printf("\nData received : ");
       for(i=0;i<7;i++)
        printf("%d",dataatrec[i]);
printf("\nCorrect message is\n");

//if errorneous bit is 0 we complement it else vice versa
if(dataatrec[7-c]==0)
dataatrec[7-c]=1;
       else
dataatrec[7-c]=0;
for (i=0;i<7;i++) {
printf("%d",dataatrec[i]);
}
}
return 0;
```

}