Assignment 5

Name: Hitesh Tolani

Roll no: 73

Class: SY-AIDS-A

Title: Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes.

Theory:

Hamming code is a powerful error-detection and error-correction technique widely employed in digital communication and data storage systems. Named after its inventor Richard Hamming, this binary linear block code adds redundant bits to the original data to detect and correct errors that may occur during transmission. The primary objective of Hamming code is to provide a reliable means of data integrity without the need for retransmission. By strategically placing redundant bits, Hamming code can identify and rectify single-bit errors, ensuring the accuracy of transmitted information. This makes it particularly valuable in applications where data integrity is crucial, such as in computer memory systems and communication protocols.

Code:

```python
data = [0] * 10
datarec = [0] * 10
print("Enter a 4 redudant bits\n")
data[7] = int(input())
data[6] = int(input())
data[5] = int(input())
data[3] = int(input())

data[4] = data[5] ^ data[6] ^ data[7]
data[2] = data[3] ^ data[6] ^ data[7]
data[1] = data[3] ^ data[5] ^ data[7]

print("Encoded data is")
for idx in range(1,8):
    print(data[idx],end='')
print("\nEnter received data bits one by one")
for idx in range(1,8):
    datarec[idx] = int(input())

c1 = datarec[1] ^ datarec[3] ^ datarec[5] ^ datarec[7];
c2 = datarec[2] ^ datarec[3] ^ datarec[6] ^ datarec[7];
```

```python
c3 = datarec[4] ^ datarec[5] ^ datarec[6] ^ datarec[7];
c = c3 * 4 + c2 * 2 + c1;
if (c == 0) :
   print("congratulations there is no error:")
else:
    print( "\error on the postion:", c);
    print("Correct message is:");
    if (datarec[c] == 0):
     datarec[c] = 1;
    else:
        datarec[c] = 0;
        for idx in range(1,8):
            print(datarec[idx])
```

Output

```
Enter a 4 redudant bits

1
1
0
1
Encoded data is
0110011
Enter received data bits one by one
1
00
1
1
0
0
1

error on the postion: 1

Correct message is:
0
0
1
1
0
0
1
                                        _
```