DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 19CS2109
COMPUTER NETWORKS AND SECURITY

ERROR CORRECTION#3

Date of the Session: ___/___/___ **Time of the Session: _____to_____**

Learning out comes:

- General idea of what Error Correction is.
- Understand and apply Hamming Code and 2-D(Multiple) Parity. [GATE-CS-2017]

IN-TUTORIAL:

1. a) ISRO is transmitting data-1011 to Chandrayan-2. For security reasons, ISRO wishes to send data by implementing the Hamming Code technique for the sake of error detection and correction (if it is one bit error). Help ISRO in performing the above

   b) A 7-bit hamming code is received as 1011101, assume Even parity and state whether received data is correct or not. If not locate the error bit?

   c) Consider a binary code that consists only four valid codewords as given below.

   00000, 01011, 10101, 11110

   Let minimum Hamming distance of code be p and maximum number of erroneous bits that can be corrected by the code be q. The value of p and q are: _____. [GATE-CS-2017]

Solution:

CNS Tutorial-3

19003187
Radhakrishna

1.a) Given message = 1011

4 bit message needs 3 redundancy bits with formula

$$2^r \geq m + r + 1$$

$r = 3$ $\quad 2^3 \geq 4 + 3 + 1$

$8 \geq 8$

∴ Redundant bits have position

$2^0, 2^1, 2^2$

Now I choose even parity

Intially message = $1\ 0\ 1\ r_3\ 1\ r_2\ r_1$

where $r_1 = 1, 3, 5, 7$

$r_1\ 1\ 1\ 1$

to get even parity $r_1 = 1$

$r_2 = 2, 3, 6, 7$

$r_2\ 1\ 0\ 1$

to get even parity $r_2 = 0$

$r_3 = 4, 5, 6, 7$

$= r_3\ 1\ 0\ 1$

to get even parity $r_3 = 0$

∴ Hamming code is 1010101

1b) Received Hamming code - 1011101

length of Hamming code - 7

redundant bits are 3 and are at positions 1, 2, 4

D_4	D_3	P_3	D_2	D_1	P_2	P_1
1	0	1	1	1	0	1

To check whether it is correct or not use following conditions (Mentioned use even parity)

r_1 = 1, 3, 5, 7

1 1 1 1 → even parity (✓) - 0

r_2 = 2, 3, 6, 7

0 1 0 1 → even parity (✓) - 0

r_3 = 4, 5, 6, 7

1 1 0 1 → odd parity (x) - 1

∴ The error is at $r_3 r_2 r_1$ - 100 - 4

∴ D_2 needs to be changed from 1 to 0

The correct Hamming code is

1010101

1.c) From given

code 1 = 00000
code 2 = 01011
code 3 = 10101
code 4 = 11110

Hamming distance is number of one's in XOR of 2 strings

00000	00000	00000
01011	10101	11110
01011 – 3	10101 – 3	11110 – 4
01011	01011	10101
10101	11110	11110
11110 – 4	10101 – 3	01011 – 3

∴ Minimum Hamming distance = 3

⇒ $p = 3$

Maximum no. of erroneous bits that can be corrected by the code we need

Hamming distance = $2d + 1$

$3 = 2d + 1$

$d = 1$

∴ $q = 1$ and $p = 3$

POST-TUTORIAL:

1. a) Data of 25 bits is arranged in 5X5 matrix (rows r0 to r4 and columns d5 to d1) and is padded with column d0 and row r5 of parity bits computed using odd party scheme. Each bit of column d0 (respectively row r5) gives the parity of the corresponding row (respectively, column). These 36 bits are transmitted over the data link.

	D5	D4	D3	D2	D1	**D0**
R0	1	1	0	0	1	**0**
R1	1	0	1	1	1	**0**
R2	1	1	1	0	0	**1**
R3	0	0	1	0	0	**0**
R4	0	1	1	1	0	**0**
R5	**0**	**0**	**0**	**1**	**0**	**0**

Check the above table and find the errors, if any. Write down the minimum possible number of corrupted bits.

b) Five packets of data, each packet containing 7 bits to be transmitted over the internet using the even parity. Append the parity bits using 2D parity with even parity, append all the parity bits to this following data. Given data is
Frame1: 1011101
Frame2: 1110111
Frame3: 1010101
Frame4: 1111011
Frame4: 1100001

Solution:

post - Tutorial

2. a)

	D_5	D_4	D_3	D_2	D_1	D_0
R_0	1	1	0	0	1	0
R_1	1	0	1	1	1	0
R_2	1	1	1	0	0	1
R_3	0	0	1	0	0	0
R_4	0	1	1	1	0	0
R_5	0	0	0	1	0	0

Given that odd parity scheme is used

R_1, R_2 and D_1, D_3 have even parity

∴ 2 rows and 2 columns have errors

⇒ Minimum no. of possible corrupted bits = 2

b) Given Frame 1 = 1011101

Frame 2 = 1110111

Frame 3 = 1010101

Frame 4 = 1111011

Frame 5 = 1100001

using Even parity

	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
R_0	1	0	1	1	1	0	1	1
R_1	1	1	1	0	1	1	1	0
R_2	1	0	1	0	1	0	1	0
R_3	1	1	1	1	0	1	1	0
R_4	1	1	0	0	0	0	1	1
R_5	1	1	0	0	1	0	1	0

2. Write a Menu driven program to generate hamming code for the given data that is given and if any hamming code is given, you have to check the errors in the code and display the bit option=int (input('Press 1 for generating hamming code in Press 2 for finding error in hamming code\n\t Enter your choice:--\n'))

Solution:

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured: _____ out of _____ Full Name of the Evaluator:

	Signature of the Evaluator Date of Evaluation:

In [2]:

```
def detectError(arr, nr):
    n = len(arr)
    res = 0
    for i in range(nr):
        val = 0
        for j in range(1, n + 1):
            if(j & (2**i) == (2**i)):
                val = val ^ int(arr[-1 * j])
        res = res + val*(10**i)
    return int(str(res), 2)
```

In [3]:

```
def calcRedundantBits(m):
    for i in range(m):
        if(2**i >= m + i + 1):
            return i
```

In [4]:

```
def posRedundantBits(data, r):
    j = 0
    k = 1
    m = len(data)
    res = ''
    for i in range(1, m + r+1):
        if(i == 2**j):
            res = res + '0'
            j += 1
        else:
            res = res + data[-1 * k]
            k += 1
    return res[::-1]
```

In [5]:

```
def calcParityBits(arr, r):
    n = len(arr)
    for i in range(r):
        val = 0
        for j in range(1, n + 1):
            if(j & (2**i) == (2**i)):
                val = val ^ int(arr[-1 * j])
        arr = arr[:n-(2**i)] + str(val) + arr[n-(2**i)+1:]
    return arr
```

In [8]:

```
while True:
  option=int (input('Press 1 for generating hamming code \nPress 2 for finding error in ham
  if option == 1:
    data = input("Enter data to be transferred:")
    m = len(data)
    r = calcRedundantBits(m)
    arr = posRedundantBits(data, r)
    arr = calcParityBits(arr, r)
    print("Data transferred is=" + arr)
  elif option == 2:
    data = input("Enter the HammingCode:")
    m = len(data)
    r = calcRedundantBits(m)
    correction = detectError(data, r)
    if correction !=0:
        print("The position of error is=",correction)
    else:
        print("No Error")
  else :
    break
```

```
Press 1 for generating hamming code
Press 2 for finding error in hamming code
Press 3 to exit  Enter your choice:--
1
Enter data to be transferred:1001
Data transferred is=1001100
Press 1 for generating hamming code
Press 2 for finding error in hamming code
Press 3 to exit  Enter your choice:--
2
Enter the HammingCode:1001101
The position of error is= 1
Press 1 for generating hamming code
Press 2 for finding error in hamming code
Press 3 to exit  Enter your choice:--
2
Enter the HammingCode:1001100
No Error
Press 1 for generating hamming code
Press 2 for finding error in hamming code
Press 3 to exit  Enter your choice:--
3
```