Bansilal Ramnath Agarwal Charitable Trust’s

# Vishwakarma Institute of Technology, Pune

(*An Autonomous Institute of Savitribai Phule Pune University*)



# Department of Artificial Intelligence and Data Science

|  |  |
| --- | --- |
| **Class/ Division** | SY-AIDS-A |
| **Name** | Archit Bagad |
| **Roll No.** | 25 |
| **GR No** | 12311951 |
| **Batch** | 03 |
| **Subject** | Computer Network |

**Lab Assignment 5**

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

**Hamming Codes:**

**Code for 4- data bit and 3 parity bits:**

#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:**

PS D:\SemesterTwo\CN\LAB Assignments> cd "d:\Second year\CN\LAB Assignments\"

; if ($?) { g++ assignment4\_1.cpp -o assignment4\_1 } ; if ($?) {

.\assignment4\_1 }

Enter 4 bits of data one by one 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:0011001

PS D:\Second year\CN\LAB Assignments>

**Code for 11- data bits and 4 parity bits:**

#include <iostream> using namespace std;

int main() {

int data[16], received[16], p1, p2, p4, p8, i, pos;

cout << "Enter 11-bit ASCII data one by one (LSB first):\n";

// Taking 11-bit ASCII input cin >> data[15];

cin >> data[14]; cin >> data[13]; cin >> data[12];

cin >> data[11]; // D7 cin >> data[10]; // D6

|  |  |
| --- | --- |
| cin >> data[9]; | // D5 |
| cin >> data[7]; | // D4 |
| cin >> data[6]; | // D3 |
| cin >> data[5]; | // D2 |
| cin >> data[3]; | // D1 |

// Calculate Parity Bits using Hamming Code data[1] = data[3] ^ data[5] ^ data[7] ^ data[9] ^

data[11]; // P1

data[2] = data[3] ^ data[6] ^ data[7] ^ data[10] ^ data[11]; // P2

data[4] = data[5] ^ data[6] ^ data[7]; // P4 data[8] = data[9] ^ data[10] ^ data[11]; // P8

cout << "\nEncoded Data (with parity bits): "; for (i = 1; i <= 15; i++)

cout << data[i];

// Simulating Transmission

cout << "\n\nEnter received data bits (one by one):\n";

for (i = 1; i <= 15; i++) cin >> received[i];

// Recalculate parity to detect errors

p1 = received[1] ^ received[3] ^ received[5] ^ received[7] ^ received[9] ^ received[11];

p2 = received[2] ^ received[3] ^ received[6] ^ received[7] ^ received[10] ^ received[11];

p4 = received[4] ^ received[5] ^ received[6] ^ received[7];

p8 = received[8] ^ received[9] ^ received[10] ^ received[11];

// Calculate error position

pos = p8 \* 8 + p4 \* 4 + p2 \* 2 + p1 \* 1;

if (pos == 0) {

cout << "\nNo error detected!";

} else {

cout << "\nError detected at position: " << pos; received[pos] = (received[pos] == 0) ? 1 : 0; //

Correct error

cout << "\nCorrected Data: "; for (i = 1; i <= 15; i++)

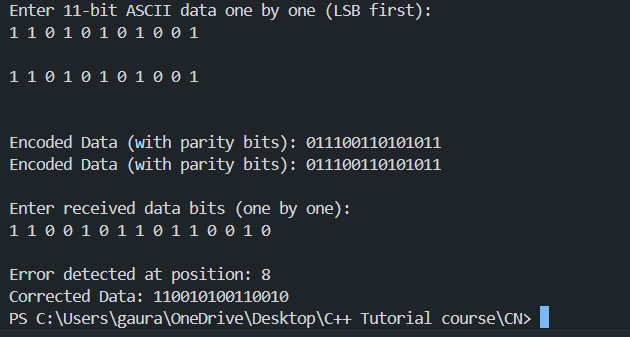
cout << received[i];

}

return 0;

}

**Output:**

****