exp: 6.

Hamming code

Aim:
Write a program to implement error detection & error correction using Hamming code concept.

Error correction with Hamming code

Sender program

* Take text input
* convert text → binary
* Apply hamming code (add redundant bits)
* Save output to channel file

Receiver program

* Read data from channel file
* check errors using Hamming code
* If error → show error position
* If no error → remove redundant bits,
* convert binary → ASCII, display text.

Program:

```
def main():
    data = list(map(int, input("Enter 4 data
                        bits (e.g., 1011): ").split()))

    d1,d2,d3,d4 = data

    P1 = d1^d2^d4

    P2 = d1^d3^d4

    P3 = d2^d3^d4

 code = [P1, P2, d1, P3, d2, d3, d4]
 print("Encoded Hamming code: ", " ".join(map(str,
                                       code)))
```

```
recv = list (map (int, input ("Enter received 7bits: ").
                                              split()))

    C1 = recv[0] ^ recv[2] ^ recv[4] ^ recv[6]
     C2 = recv[1] ^ recv[2] ^ recv[5] ^ recv[6]
     C3 = recv[3] ^ recv[4] ^ recv[5] ^ recv[6]

   errorPos = C1 + (C2 << 1) + (C3 << 2)

  if errorpos == 0 :
       print ('No error detected')
  else:
       Print ("error at bit position:", errorPos)

       recv [errorPos - 1] ^= 1
       print ("corrected code:", " ".join (map (str, recv)))

if __name__ == "__main__":
       main ()
```

Result:

Hence the required program for error detection & error correction is written & executed successfully.

Sample Input output:

```
Enter 4 data bits : 1011
Encoded Hamming code : 0110011
Enter received 7 bits : 0111011
Error at bit position : 4
corrected code : 0110011
```

[illegible] 18/9/25 10/10