Program:

import math  
  
  
def count\_odd\_even(parity\_list, even\_or\_odd):  
 count = 0  
 bit = **"1"** if even\_or\_odd == **"even"**:  
 for i in range(len(parity\_list)):  
 if parity\_list[i] == bit:  
 count += 1  
 if count % 2 == 0:  
 return 0  
 else:  
 return 1  
 if even\_or\_odd == **"odd"**:  
 for i in range(len(parity\_list)):  
 if parity\_list[i] == bit:  
 count += 1  
 if count % 2 != 0:  
 return 0  
 else:  
 return 1  
  
  
def parity(message\_bits, parity\_check,even\_or\_odd):  
 parity\_list = []  
 error\_word = []  
 if even\_or\_odd == **"even"**:  
 print(**"**\n**For Even Parity"**)  
 else:  
 print(**"**\n**For Odd Parity"**)  
 for i in range(int(parity\_check)):  
 start = int(pow(2,i))-1  
 message\_bits.insert(start,0)  
 for i in range(int(parity\_check)):  
 start = int(pow(2, i))  
 skip = start \* 2  
 z = len(message\_bits)  
 if skip >= 8:  
 z = len(message\_bits) + 1  
 for j in range(start):  
 for k in range(start, z, skip):  
 a = k - 1  
 parity\_list.append(message\_bits[a])  
 start += 1  
 parity\_list.pop(0)  
 error\_word.append(count\_odd\_even(parity\_list, even\_or\_odd))  
 parity\_list = []  
 print(**"Parity Bit "** + str(i+1) + **" = "** + str(error\_word[i]))  
 for x in range(int(parity\_check)):  
 start = int(pow(2, x)) - 1  
 message\_bits[start] = error\_word[x]  
 print(**"Therefore the Codeword = "**+**''**.join([str(elem) for elem in message\_bits]))  
  
  
def exit\_code():  
 print(**"**\n**Program ended successfully"**)  
 exit()  
  
  
def hamming():  
 temp = input(**"Enter the message bits : "**)  
 message\_bits = [str(x) for x in temp]  
 parity\_check = int(math.log2(len(message\_bits)) + 1)  
 print(**"Therefore, there will be "** + str(parity\_check) + **" parity/check bits in codeword"**)  
 while True:  
 print(**"**\n**Press 1 for Even Parity"**)  
 print(**"Press 2 for Odd Parity"**)  
 print(**"Press 3 t exit"**)  
 choice = int(input(**"Enter your choice : "**))  
 if choice == 1:  
 parity(message\_bits, int(parity\_check),**"even"**)  
 if choice == 2:  
 parity(message\_bits, int(parity\_check),**"odd"**)  
 if choice == 3:  
 exit\_code()  
  
  
print(**"HAMMING CODE TECHNIQUE**\n**"**)  
hamming()

Output:

HAMMING CODE TECHNIQUE

Enter the message bits : 1101001100110101

Therefore, there will be 5 parity/check bits in codeword

Press 1 for Even Parity

Press 2 for Odd Parity

Press 3 t exit

Enter your choice : 1

For Even Parity

Parity Bit 1 = 0

Parity Bit 2 = 1

Parity Bit 3 = 1

Parity Bit 4 = 1

Parity Bit 5 = 1

Therefore the Codeword = 011110110011001110101

Press 1 for Even Parity

Press 2 for Odd Parity

Press 3 t exit

Enter your choice :

**Conclusion**: The Hamming Code technique was studied, successfully implemented and appropriate output was obtained.