

Assinment No :

Hamming Code:

```
#include <stdio.h>

int main() {
    int data[11]; // Array to hold encoded data
    int datacheck[11]; // Array to hold received data for checking
    int c1, c2, c3, c4, c, i;

    // Input 7 bits of data
    printf("Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):\n");
    scanf("%d", &data[2]); // D1
    scanf("%d", &data[4]); // D2
    scanf("%d", &data[5]); // D3
    scanf("%d", &data[6]); // D4
    scanf("%d", &data[8]); // D5
    scanf("%d", &data[9]); // D6
    scanf("%d", &data[10]); // D7

    // Calculate parity bits
    data[0] = data[2] ^ data[4] ^ data[6] ^ data[8] ^ data[10]; // P1
    data[1] = data[2] ^ data[5] ^ data[6] ^ data[9] ^ data[10]; // P2
    data[3] = data[4] ^ data[5] ^ data[6] ^ data[10]; // P4
    data[7] = data[8] ^ data[9] ^ data[10]; // P8

    // Display encoded data
    printf("\nEncoded data is:\n");
```

```

for (i = 0; i < 11; i++)
    printf("%d", data[i]);
printf("\n");

// Input received data
printf("\nEnter received data bits one by one:\n");
for (i = 0; i < 11; i++)
    scanf("%d", &datacheck[i]);

// Compute parity check bits
c1 = datacheck[0] ^ datacheck[2] ^ datacheck[4] ^ datacheck[6] ^
datacheck[8] ^ datacheck[10]; // P1 check

c2 = datacheck[1] ^ datacheck[2] ^ datacheck[5] ^ datacheck[6] ^
datacheck[9] ^ datacheck[10]; // P2 check

c3 = datacheck[3] ^ datacheck[4] ^ datacheck[5] ^ datacheck[6] ^
datacheck[10]; // P4 check

c4 = datacheck[7] ^ datacheck[8] ^ datacheck[9] ^ datacheck[10]; // P8 check

c = c4 * 8 + c3 * 4 + c2 * 2 + c1; // Error position

if (c == 0) {
    printf("\nNo error in the transmission of data.\n");
} else {
    printf("\nFound error at position %d\n", c);

// Display received and encoded data
printf("\nEncoded data: ");
for (i = 0; i < 11; i++)

```

```
    printf("%d", data[i]);\n\n    printf("\nReceived data: ");\n    for (i = 0; i < 11; i++)\n        printf("%d", datacheck[i]);\n\n    // Correct the erroneous bit\n    if (datacheck[11 - c] == 0)\n        datacheck[11 - c] = 1;\n    else\n        datacheck[11 - c] = 0;\n\n    // Display corrected data\n    printf("\nCorrected message is: ");\n    for (i = 0; i < 11; i++)\n        printf("%d", datacheck[i]);\n    printf("\n");\n}\n\nreturn 0;\n}
```

Output Without Error :

The screenshot shows a C programming interface on a Windows desktop. The code in main.c reads 7 bits of data one by one, calculates parity bits, and displays the encoded data. The output window shows the input data, the calculated encoded data, and the received data bits. A sidebar for Programiz PRO offers premium courses.

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int data[11]; // Array to hold encoded data
5     int datacheck[11]; // Array to hold received data for checking
6     int c1, c2, c3, c4, c, i;
7
8     // Input 7 bits of data
9     printf("Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):\n");
10    scanf("%d", &data[2]); // D1
11    scanf("%d", &data[4]); // D2
12    scanf("%d", &data[5]); // D3
13    scanf("%d", &data[6]); // D4
14    scanf("%d", &data[8]); // D5
15    scanf("%d", &data[9]); // D6
16    scanf("%d", &data[10]); // D7
17
18    // Calculate parity bits
19    data[0] = data[2] ^ data[4] ^ data[6] ^ data[8] ^ data[10]; // P1
20    data[1] = data[2] ^ data[5] ^ data[6] ^ data[9] ^ data[10]; // P2
21    data[3] = data[4] ^ data[5] ^ data[6] ^ data[10]; // P4
22    data[7] = data[8] ^ data[9] ^ data[10]; // P8
23
24    // Display encoded data
25    printf("\nEncoded data is:\n");
```

Output

```
Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):
1
0
1
1
0
0
1
Encoded data is:
10110111001
Enter received data bits one by one:
1
0
1
1
0
1
0
1
No error in the transmission of data.
```

Programiz PRO
Premium Courses by Programiz
Learn More

Output With Error :

The screenshot shows a C programming interface on a Windows desktop. The code is identical to the previous one but with a different input. The output window shows the input data, the calculated encoded data, and the received data bits. An error message indicates a discrepancy at position 1.

```
main.c
1 #include <stdio.h>
2
3 int main() {
4     int data[11]; // Array to hold encoded data
5     int datacheck[11]; // Array to hold received data for checking
6     int c1, c2, c3, c4, c, i;
7
8     // Input 7 bits of data
9     printf("Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):\n");
10    scanf("%d", &data[2]); // D1
11    scanf("%d", &data[4]); // D2
12    scanf("%d", &data[5]); // D3
13    scanf("%d", &data[6]); // D4
14    scanf("%d", &data[8]); // D5
15    scanf("%d", &data[9]); // D6
16    scanf("%d", &data[10]); // D7
17
18    // Calculate parity bits
19    data[0] = data[2] ^ data[4] ^ data[6] ^ data[8] ^ data[10]; // P1
20    data[1] = data[2] ^ data[5] ^ data[6] ^ data[9] ^ data[10]; // P2
21    data[3] = data[4] ^ data[5] ^ data[6] ^ data[10]; // P4
22    data[7] = data[8] ^ data[9] ^ data[10]; // P8
23
24    // Display encoded data
25    printf("\nEncoded data is:\n");
```

Output

```
Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):
1
0
1
1
0
0
1
Encoded data is:
11110010110
Enter received data bits one by one:
0
1
0
1
1
1
0
1
Found error at position 1
```

Programiz PRO
Premium Courses by Programiz
Learn More