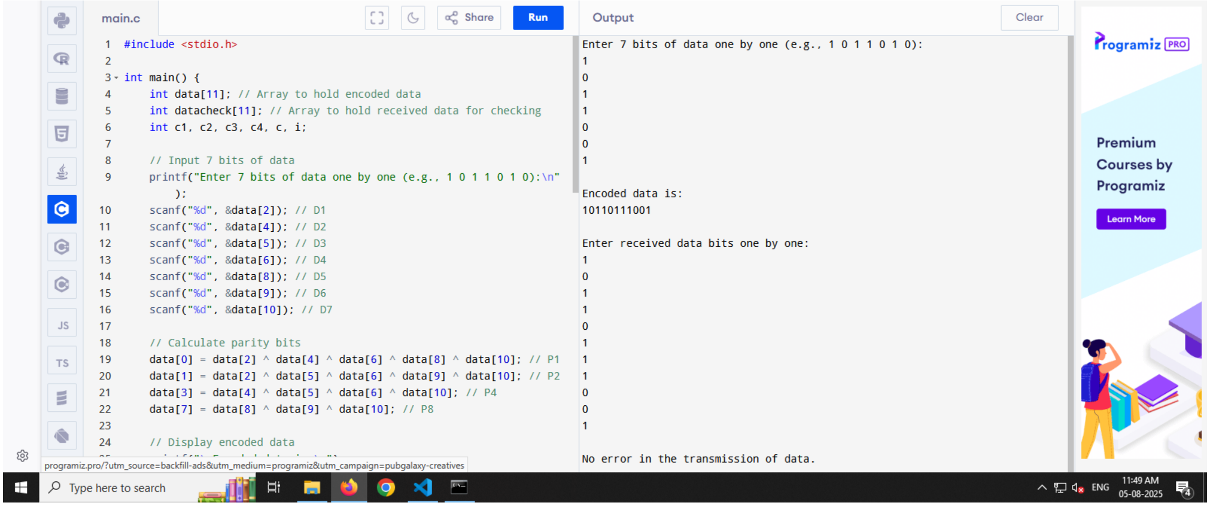
Assinment No :

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]);

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

        // Correct the erroneous bit
        if (datacheck[11 - c] == 0)
            datacheck[11 - c] = 1;
        else
            datacheck[11 - c] = 0;

        // Display corrected data
        printf("\nCorrected message is: ");
        for (i = 0; i < 11; i++)
            printf("%d", datacheck[i]);
        printf("\n");
    }

    return 0;
}
```

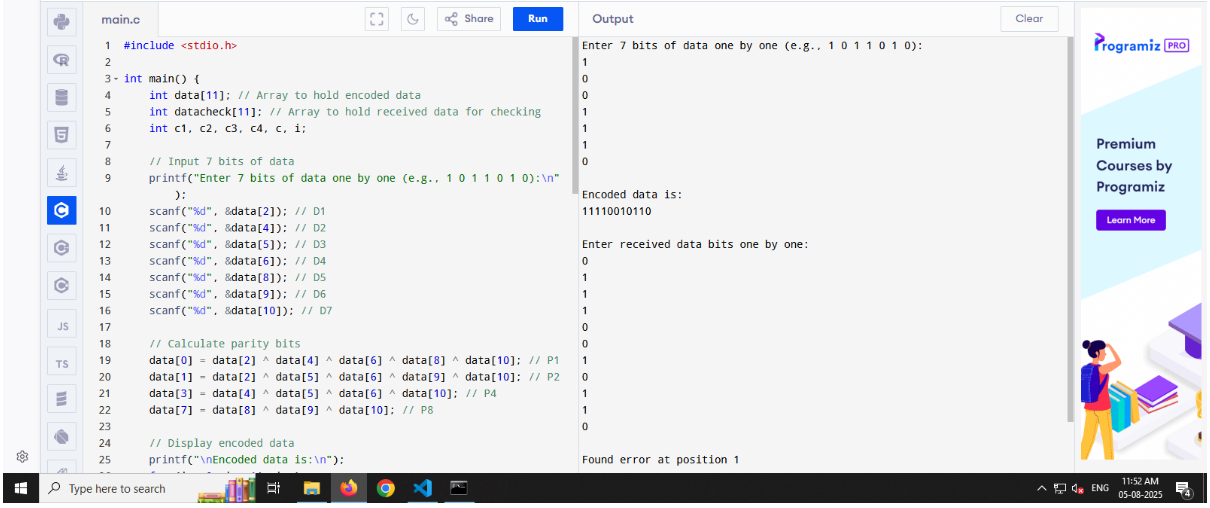
Output Without Error :


main.c
Share
Run
Output
Clear
Programiz PRO
Premium Courses by Programiz
Learn More
programiz.pro/?utm_source=backfill-ads&utm_medium=programiz&utm_campaign=pubgalaxy-creatives
Type here to search
ENG
11:49 AM
05-08-2025

```c
#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
```

```
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
1
1
0
0
1

No error in the transmission of data.
```

Output With Error :



```c
#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");
```

```
Enter 7 bits of data one by one (e.g., 1 0 1 1 0 1 0):
1
0
0
1
1
1
0

Encoded data is:
11110010110

Enter received data bits one by one:
0
1
1
1
0
0
1
0
1
1
0

Found error at position 1
```