EXPERIMENT-4

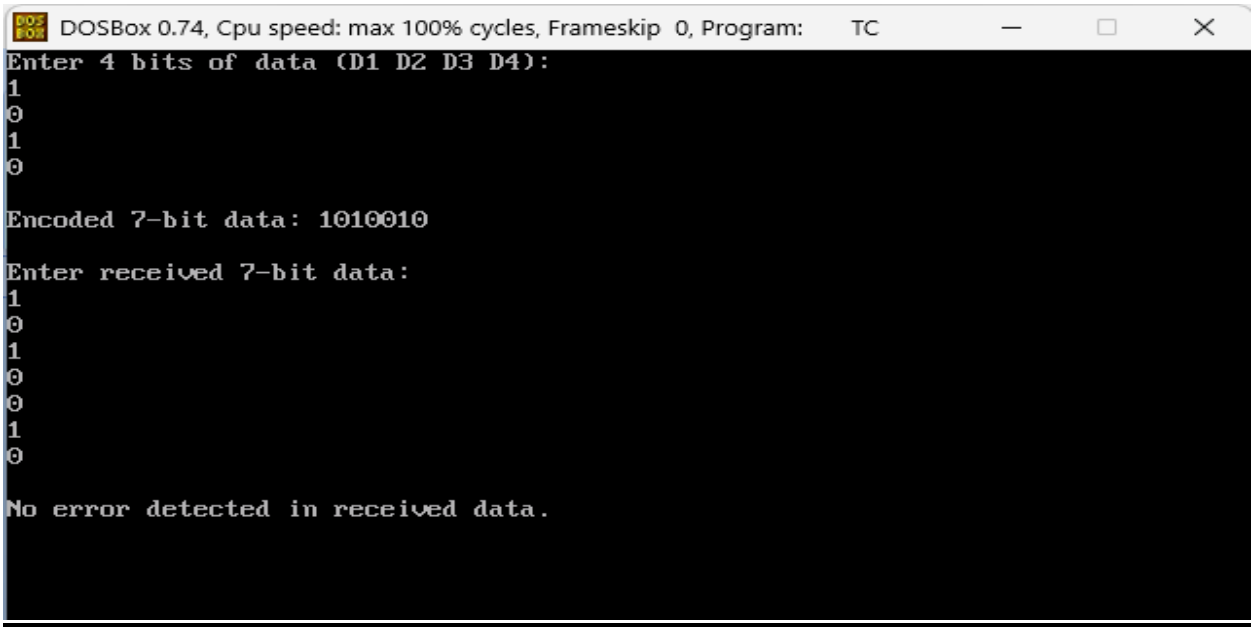
Write a program for Hamming Code generation for error detection and correction.

Program

```
#include <stdio.h>
void main()
{
   int data[8];          // data[1] to data[7]
   int received[8];
   int r, r1, r2, r4, i;
   clrscr();
   printf("Enter 4 bits of data (D1 D2 D3 D4):\n");
   scanf("%d", &data[7]); // D1
   scanf("%d", &data[6]); // D2
   scanf("%d", &data[5]); // D3
   scanf("%d", &data[3]); // D4
   // Calculate parity bits (even parity)
   data[1] = data[3] ^ data[5] ^ data[7]; // P1
   data[2] = data[3] ^ data[6] ^ data[7]; // P2
   data[4] = data[5] ^ data[6] ^ data[7]; // P4
   printf("\nEncoded 7-bit data: ");
   for (i = 7; i >= 1; i--)
         printf("%d", data[i]);
   // Receiving data
   printf("\n\nEnter received 7-bit data:\n");
   for (i = 7; i >= 1; i--)
         scanf("%d", &received[i]);
   // Syndrome bits
   r1 = received[1] ^ received[3] ^ received[5] ^ received[7];
   r2 = received[2] ^ received[3] ^ received[6] ^ received[7];
   r4 = received[4] ^ received[5] ^ received[6] ^ received[7];
   r = r4 * 4 + r2 * 2 + r1 * 1;
   if (r == 0)
   {
         printf("\nNo error detected in received data.\n");
   }
   else
   {
         printf("\nError detected at position: %d\n",r);
         printf("Received data : ");
         for (i = 7; i >= 1; i--)
            printf("%d", received[i]);
         // Correct the error
```

```
        received[r] =received[r] ^ 1;
        printf("\nCorrected data: ");
        for (i = 7; i >= 1; i--)
            printf("%d", received[i]);
        printf("\n");
    }
    getch();
}
```

Output 1:



Output 2: