#include <stdio.h>

int main() {

int data[11] = {0}; // Encoded 11-bit data with parity and data bits

int datacheck[11]; // Received 11-bit data to check for errors

int c, c1, c2, c3, c4, i;

printf("Enter 7 bits of data one by one.:\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];

data[1] = data[2] ^ data[5] ^ data[6] ^ data[9] ^ data[10];

data[3] = data[4] ^ data[5] ^ data[6];

data[7] = data[8] ^ data[9] ^ data[10];

printf("\nEncoded 11-bit Hamming code:\n");

for(i = 0; i < 11; i++)

printf("%d", data[i]);

printf("\n");

printf("\nEnter received 11-bit data one by one:\n");

for(i = 0; i < 11; i++)

scanf("%d", &datacheck[i]);

// Syndrome bits (error detection)

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]; // P4 check

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

c = c4 \* 8 + c3 \* 4 + c2 \* 2 + c1 \* 1;

if(c == 0) {

printf("\nNo error detected during transmission.\n");

} else {

printf("\nError detected at bit position: %d (1-based)\n", c);

printf("Data before correction: ");

for(i = 0; i < 11; i++)

printf("%d", datacheck[i]);

datacheck[c - 1] = !datacheck[c - 1];

printf("\nCorrected data: ");

for(i = 0; i < 11; i++)

printf("%d", datacheck[i]);

printf("\n");

}

return 0;

}