```
/*CRC*/
#include <stdio.h>
#include <stdbool.h>

// Function prototypes
bool crcCheck(int message[], int n, int polynomial[], int m);

int main() {
    int message[100], polynomial[100];
    int message_length, polynomial_length;

    printf("Enter length of message: ");
    scanf("%d", &message_length);

    printf("Enter message bits (0s and 1s): ");
    for (int i = 0; i < message_length; i++)
        scanf("%d", &message[i]);

    printf("Enter length of polynomial: ");
    scanf("%d", &polynomial_length);

    printf("Enter polynomial coefficients (0s and 1s): ");
    for (int i = 0; i < polynomial_length; i++)
        scanf("%d", &polynomial[i]);

    // Append zeros (padding) to message for polynomial length - 1 times
    for (int i = 0; i < polynomial_length - 1; i++)
        message[message_length + i] = 0;

    printf("\nMessage after appending zeros for CRC: ");
    for (int i = 0; i < message_length + polynomial_length - 1; i++)
```

```
        printf("%d ", message[i]);
    printf("\n");

    if (crcCheck(message, message_length + polynomial_length - 1, polynomial, polynomial_length))
        printf("\nCRC check passed. No error detected.\n");
    else
        printf("\nCRC check failed. Error detected.\n");

    return 0;
}

// Function to perform CRC check
bool crcCheck(int message[], int n, int polynomial[], int m) {
    for (int i = 0; i <= n - m; i++) {
        if (message[i] == 1) {
            for (int j = 0; j < m; j++) {
                message[i + j] = message[i + j] ^ polynomial[j];
            }
        }
    }

    // Check if remainder is all zeros
    for (int i = n - m + 1; i < n; i++) {
        if (message[i] != 0)
            return false;
    }

    return true;
}
```

```
Enter length of message: 10
Enter message bits (0s and 1s): 1 0 1 0 1 0 1 0 1 0
Enter length of polynomial: 5
Enter polynomial coefficients (0s and 1s): 1 1 0 0 1

Message after appending zeros for CRC: 1 0 1 0 1 0 1 0 1 0 0 0 0 0

CRC check failed. Error detected.

--------------------------------
Process exited after 27.61 seconds with return value 0
Press any key to continue . . .
```