Cycle-II

Program 1

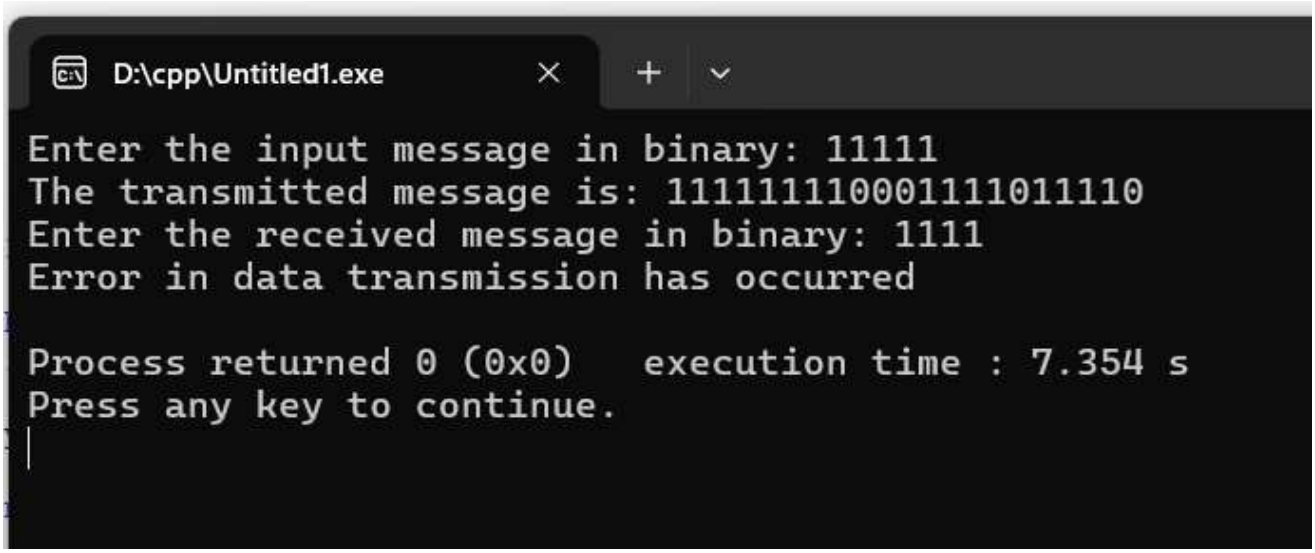
i. Write a program for error detecting code using CRC-CCITT (16-bits).

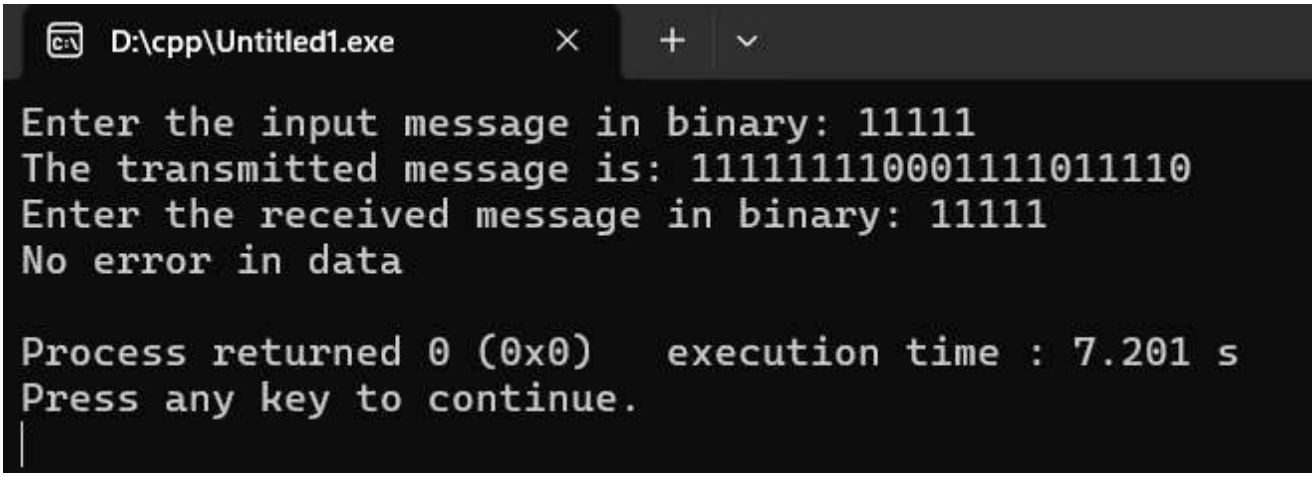
ii. Procedure

```
#include <stdio.h>
#include <string.h>

int crc(char *ip, char *op, char *poly, int mode) {
    strcpy(op, ip);

    if (mode) {
        for (int i = 1; i < strlen(poly); i++) {
            strcat(op, "0");
        }
    }

    // Perform XOR on the message with the selected polynomial
    for (int i = 0; i < strlen(ip); i++) {
        if (op[i] == '1') {
            for (int j = 0; j < strlen(poly); j++) {
                if (op[i + j] == poly[j])
                    op[i + j] = '0';
                else
                    op[i + j] = '1';
            }
        }
    }

    // Check for errors. Return 0 if error detected
    for (int i = 0; i < strlen(op); i++) {
        if (op[i] == '1')
            return 0;
    }

    return 1;
}

int main() {
    char ip[50], op[50], recv[50];
    char poly[] = "10001000000100001";

    printf("Enter the input message in binary: ");
    scanf("%s", ip);
```

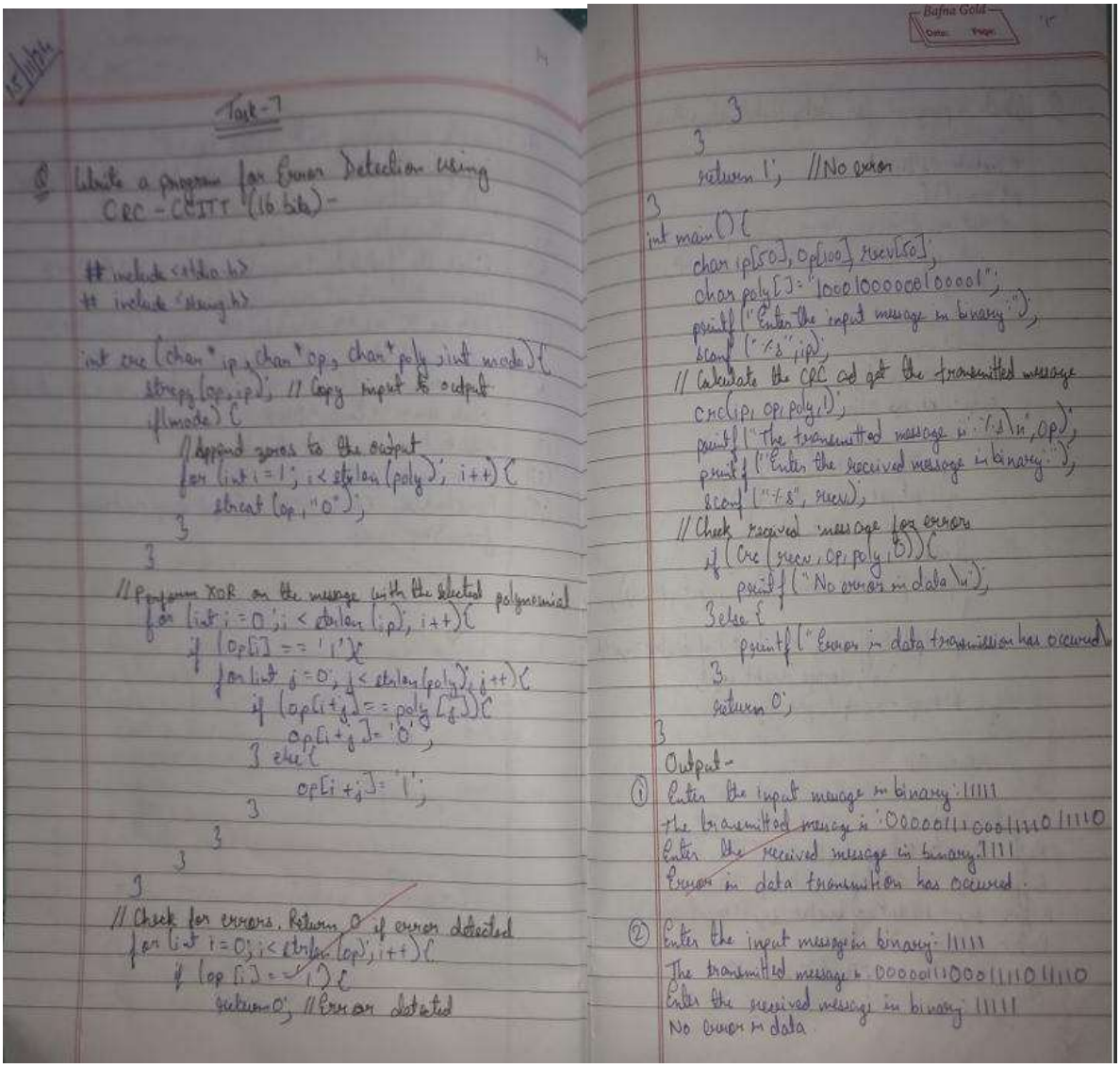
```
    crc(ip, op, poly, 1);

    printf("The transmitted message is: %s%s\n", ip, op + strlen(ip));

    printf("Enter the received message in binary: ");
    scanf("%s", recv);

    if (crc(recv, op, poly, 0)) {
        printf("No error in data\n");
    } else {
        printf("Error in data transmission has occurred\n");
    }

    return 0;
}
```

iii. Screen shots/ output





iv. Observation



Task-7

Q. Write a program for Error Detection using CRC-CCITT (16 bits)-

```
#include<stdio.h>
#include<string.h>

int crc(char *ip, char *op, char *poly, int mode){
    strcpy(op, ip); // Copy input to output
    if(mode){
        // Append zeros to the output
        for(int i=1; i<strlen(poly); i++){
            strcat(op, "0");
        }
    }
    // Perform XOR on the message with the selected polynomial
    for(int i=0; i<strlen(ip); i++){
        if(op[i]=='1'){
            for(int j=0; j<strlen(poly); j++){
                if(op[i+j]==poly[j]){
                    op[i+j]='0';
                }else{
                    op[i+j]='1';
                }
            }
        }
    }
    // Check for errors. Return 0 if error detected
    for(int i=0; i<strlen(op); i++){
        if(op[i]=='1'){
            return 0; // Error detected
        }
    }
    return 1; // No error
}
int main(){
    char ip[50], op[100], recv[50];
    char poly[]="1000100000010001";
    printf("Enter the input message in binary:");
    scanf("%s",ip);
    // Calculate the CRC and get the transmitted message
    crc(ip, op, poly, 1);
    printf("The transmitted message is:%s\n",op);
    printf("Enter the received message in binary:");
    scanf("%s", recv);
    // Check received message for errors
    if(crc(recv, op, poly, 0)){
        printf("No errors in data\n");
    }else{
        printf("Error in data transmission has occured.\n");
    }
    return 0;
}
```

Output-

① Enter the input message in binary: 11111
The transmitted message is: 0000011100011110 11110
Enter the received message in binary: 1111
Error in data transmission has occured.

② Enter the input message in binary: 11111
The transmitted message is: 0000011100011110 11110
Enter the received message in binary: 11111
No errors in data