Error detection code using crc ccitt

```
#include <stdio.h>
#include <stdint.h>
#define CRC POLY 0x11021 // Polynomial for CRC-CCITT (x^16 + x^12 + x^5 + 1)
#define INITIAL_CRC 0xFFFF // Initial value of CRC for CRC-CCITT
// Function to compute the CRC for a given data buffer
uint16_t compute_crc(uint8_t *data, size_t length) {
  uint16_t crc = INITIAL_CRC;
  // Process each byte in the input data
  for (size t i = 0; i < length; i++) {
     crc ^= (data[i] << 8); // Move byte into upper byte of CRC
     // Process each bit of the byte
     for (int j = 0; j < 8; j++) {
       if (crc & 0x8000) {
          crc = (crc << 1) ^ CRC POLY; // Shift left and apply the polynomial if the leftmost bit is 1
       } else {
          crc <<= 1; // Otherwise just shift left
    }
  }
  return crc & 0xFFFF; // Return the CRC (ensure it's 16 bits)
// Function to check if the received data is valid
int check_crc(uint8_t *data, size_t length, uint16_t expected crc) {
  uint16_t computed_crc = compute_crc(data, length);
  return (computed crc == expected crc); // Return 1 if CRC matches, else 0
}
// Main function to demonstrate CRC-CCITT
int main() {
  uint8 t data[] = "Hello, World!"; // Example data
  size_t data_length = sizeof(data) - 1; // Exclude the null terminator
  printf("Data: %s\n", data);
  // Compute the CRC for the data
  uint16_t crc = compute_crc(data, data_length);
  // Display the computed CRC
  printf("Computed CRC-CCITT: 0x%04X\n", crc);
  // Simulate a transmission and check for errors by comparing CRC
  uint8 t received data[] = "Hello, World!";
  size t received length = sizeof(received data) - 1;
  // Check if the received data has the correct CRC
  if (check crc(received data, received length, crc)) {
     printf("Data received correctly with no errors.\n");
  } else {
     printf("Error detected in received data!\n");
  return 0;
```

Output:

```
Data: Hello, World!
Computed CRC-CCITT: 0x67DA
Data received correctly with no errors.

...Program finished with exit code 0
Press ENTER to exit console.
```

A 350	classmate
17/12/24	Date
	write a program for error detection water using
	CTC CCITE
	thing wints t data(7: "Hillo portd")
	#include & stdirt-his
	"(white "a/s/ solog") strict comments
	# deline CDC DOLV DVIII
Catoria	# define CRC-POLY OX11021 # define PNITIAL CRC OXFFFF
16193	* OLD ON XO: TILD DE Betugrache) Pring
(1)	uint16_t compute_crc(uint8_t + data, singe_t length)
	Ebroa other a stab beness of 31/10
· c. (6) 5) b	- by 130 re wint to L+ CRO' = INIT IN WICRO; + 100
1	for (i=0 : i < ungth : i++)2
ate	al borison MCRCA Totalatij < (8);)
	133
2	for (Int; =0;j28;j++)?
on atic	10 (CRCBOX 8000) 21
	(CACRC = (CRC XXI)^ CAC-POLY
	3
	elae 2 S sels
F	visione of best CRC 2xx 2°;] thore
	3 ("n(lata) data). E
	3
	3 Charles of the second of the
	f l
	int check_crc (uint 8_t * data, size_tlength, wint 16_t expected_crc)?
	uintib_t computed cre = compute - cre (data,
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	AOTO XO () 4) boughas (MTh)
	0x89 and Afic schum (computed cpc = = expected_cpc);
	3 14.
	11/0/
	23

