

Assignment-3

Mohammed Nizamuddin

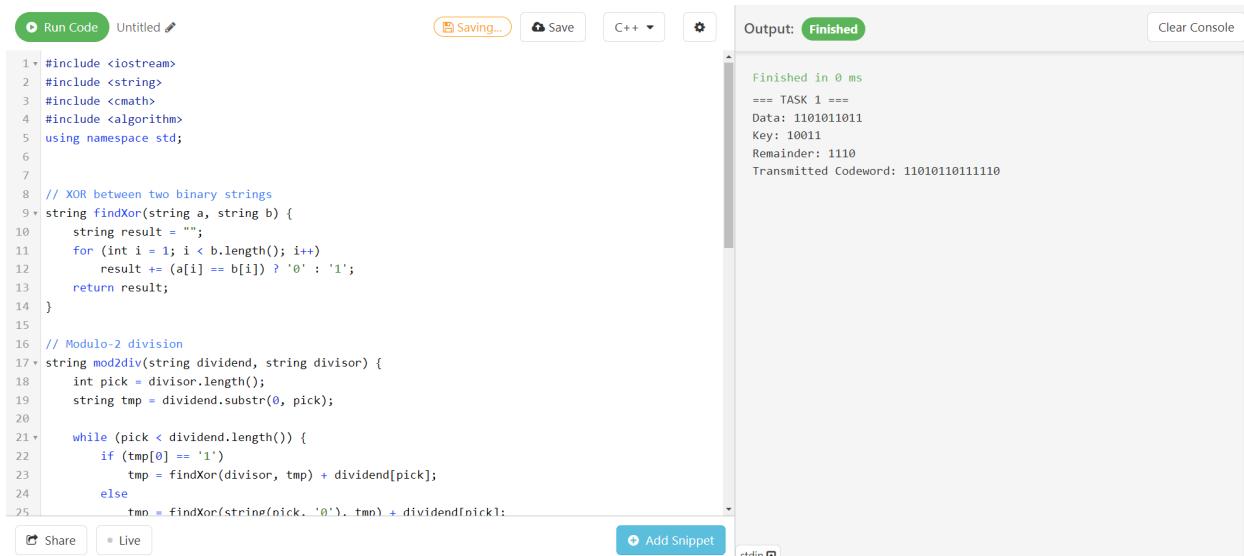
2503B05144

Task:1

Problem-01:

-

A bit stream 1101011011 is transmitted using the standard CRC method.
The generator polynomial is x^4+x+1 . What is the actual bit string transmitted?



The screenshot shows a code editor interface with the following details:

- Run Code** button (green)
- Untitled file
- Saving...** status
- Save** button
- C++** dropdown
- Settings** icon
- Output:** **Finished**
- Clear Console** button

```
1 // include <iostream>
2 #include <string>
3 #include <cmath>
4 #include <algorithm>
5 using namespace std;
6
7
8 // XOR between two binary strings
9 string findXor(string a, string b) {
10     string result = "";
11     for (int i = 1; i < b.length(); i++)
12         result += (a[i] == b[i]) ? '0' : '1';
13     return result;
14 }
15
16 // Modulo-2 division
17 string mod2div(string dividend, string divisor) {
18     int pick = divisor.length();
19     string tmp = dividend.substr(0, pick);
20
21     while (pick < dividend.length()) {
22         if (tmp[0] == '1')
23             tmp = findXor(divisor, tmp) + dividend[pick];
24         else
25             tmp = findXor(string(pick - '0'), tmp) + dividend[pick];
26     }
27     return tmp;
28 }
```

Output:

```
Finished in 0 ms
*** TASK 1 ===
Data: 1101011011
Key: 10011
Remainder: 1110
Transmitted Codeword: 11010110111110
```

Bottom Buttons:

- Share**
- Live**
- Add Snippet**

The screenshot shows a C++ development environment with the following details:

- Code Editor:** The code is written in C++ and defines a function `encodeData` and a `main` function. The `encodeData` function appends a key-length zero-padded string to the data, performs modular division, and returns the encoded data. The `main` function initializes data and key, prints them, encodes the data, and prints the transmitted codeword.
- Output Panel:** The output is labeled "Finished" and shows the following text:
 - Finished in 0 ms
 - ==== TASK 1 ===
 - Data: 1101011011
 - Key: 10011
 - Remainder: 1110
 - Transmitted Codeword: 11010110111110
- Toolbar:** Includes "Run Code", "Save", "C++", and settings icons.
- Status Bar:** Shows "Output: Finished".

Task:2

Problem-02:

A bit stream 10011101 is transmitted using the standard CRC method. The generator polynomial is x^3+1 .

1. What is the actual bit string transmitted?
2. Suppose the third bit from the left is inverted during transmission.
How will receiver detect this error?

Run Code Untitled

```

1 #include <iostream>
2 #include <string>
3 #include <cmath>
4 #include <algorithm>
5 using namespace std;
6 // XOR between two binary strings
7 string findXor(string a, string b) {
8     string result = "";
9     for (int i = 1; i < b.length(); i++)
10         result += (a[i] == b[i]) ? '0' : '1';
11     return result;
12 }
13 // Modulo-2 division
14 string mod2div(string dividend, string divisor) {
15     int pick = divisor.length();
16     string tmp = dividend.substr(0, pick);
17
18     while (pick < dividend.length()) {
19         if (tmp[0] == '1')
20             tmp = findXor(divisor, tmp) + dividend[pick];
21         else
22             tmp = findXor(string(pick, '0'), tmp) + dividend[pick];
23         pick++;
24     }
25     if (tmp[0] == '1')

```

Output: Finished

```

Finished in 0 ms
== TASK 2 ==
Data: 10011101
Key: 1001
Remainder: 100
Transmitted Codeword: 10011101100
Received (error at bit 3): 10111101100
Receiver: Error Detected

```

Run Code Untitled

```

24
25     }
26     if (tmp[0] == '1')
27         tmp = findXor(divisor, tmp);
28     else
29         tmp = findXor(string(pick, '0'), tmp);
30     return tmp;
31 }
32 // Encode data with CRC
33 string encodeData(string data, string key) {
34     int keyLen = key.length();
35     string appendedData = data + string(keyLen - 1, '0');
36     string remainder = mod2div(appendedData, key);
37     cout << "Remainder: " << remainder << endl;
38     return data + remainder;
39 }
40 // Receiver check
41 bool receiverCheck(string codeword, string key) {
42     string remainder = mod2div(codeword, key);
43     return (remainder.find('1') == string::npos);
44 }
45 int main() {
46     string data = "10011101";
47     string key = "1001"; // x^3 + 1
48     cout << "== TASK 2 ==" << endl;
49     cout << "Data: " << data << endl;

```

Output: Finished

```

Finished in 0 ms
== TASK 2 ==
Data: 10011101
Key: 1001
Remainder: 100
Transmitted Codeword: 10011101100
Received (error at bit 3): 10111101100
Receiver: Error Detected

```

Run Code Untitled

```

39 // Receiver check
40 bool receiverCheck(string codeword, string key) {
41     string remainder = mod2div(codeword, key);
42     return (remainder.find('1') == string::npos);
43 }
44 int main() {
45     string data = "10011101";
46     string key = "1001"; // x^3 + 1
47     cout << "== TASK 2 ==" << endl;
48     cout << "Data: " << data << endl;
49     cout << "Key: " << key << endl;
50     string codeword = encodeData(data, key);
51     cout << "Transmitted Codeword: " << codeword << endl;
52     // Introduce error at 3rd bit
53     string received = codeword;
54     received[2] = (received[2] == '0') ? '1' : '0';
55     cout << "Received (error at bit 3): " << received << endl;
56
57     if (receiverCheck(received, key))
58         cout << "Receiver: No Error Detected" << endl;
59     else
60         cout << "Receiver: Error Detected" << endl;
61     return 0;
62 }
63

```

Output: Finished

```

Finished in 0 ms
== TASK 2 ==
Data: 10011101
Key: 1001
Remainder: 100
Transmitted Codeword: 10011101100
Received (error at bit 3): 10111101100
Receiver: Error Detected

```