Assignment-3

Mohammed Nizamuddin

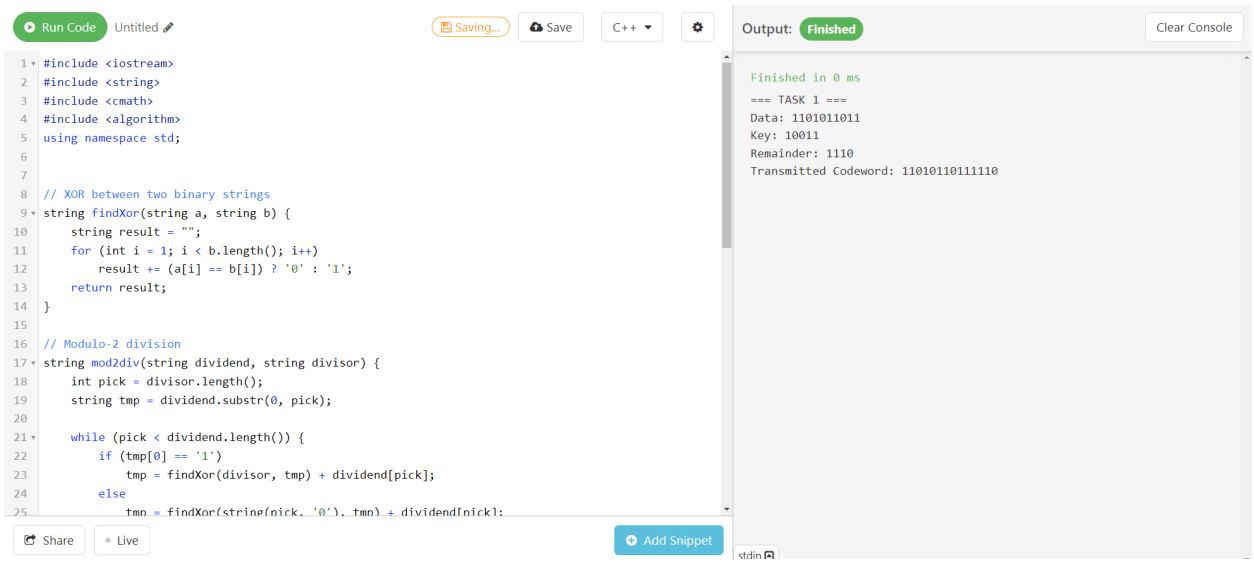
2503B05144

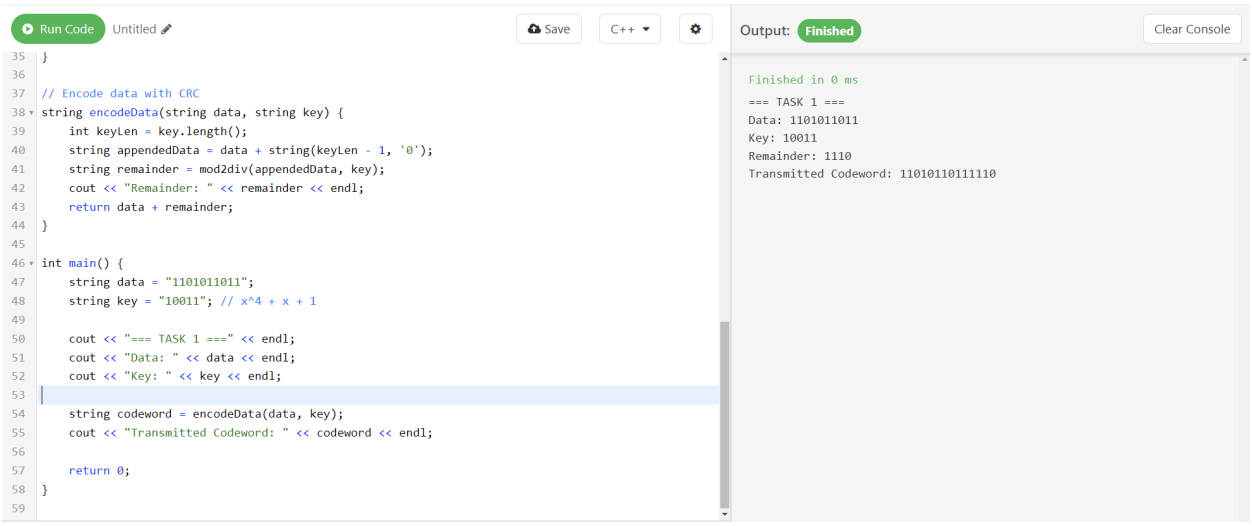
Task:1

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?





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 | Save | C++

```cpp
#include <iostream>
#include <string>
#include <cmath>
#include <algorithm>
using namespace std;
// XOR between two binary strings
string findXor(string a, string b) {
    string result = "";
    for (int i = 1; i < b.length(); i++)
        result += (a[i] == b[i]) ? '0' : '1';
    return result;
}
// Modulo-2 division
string mod2div(string dividend, string divisor) {
    int pick = divisor.length();
    string tmp = dividend.substr(0, pick);

    while (pick < dividend.length()) {
        if (tmp[0] == '1')
            tmp = findXor(divisor, tmp) + dividend[pick];
        else
            tmp = findXor(string(pick, '0'), tmp) + dividend[pick];
        pick++;
    }
    if (tmp[0] == '1')
```

Output: Finished | Clear Console

```
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 | Save | C++

```cpp
    }
    if (tmp[0] == '1')
        tmp = findXor(divisor, tmp);
    else
        tmp = findXor(string(pick, '0'), tmp);
    return tmp;
}
// Encode data with CRC
string encodeData(string data, string key) {
    int keyLen = key.length();
    string appendedData = data + string(keyLen - 1, '0');
    string remainder = mod2div(appendedData, key);
    cout << "Remainder: " << remainder << endl;
    return data + remainder;
}
// Receiver check
bool receiverCheck(string codeword, string key) {
    string remainder = mod2div(codeword, key);
    return (remainder.find('1') == string::npos);
}
int main() {
    string data = "10011101";
    string key = "1001"; // x^3 + 1
    cout << "=== TASK 2 ===" << endl;
```

Output: Finished | Clear Console

```
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 | Save | C++

```cpp
// Receiver check
bool receiverCheck(string codeword, string key) {
    string remainder = mod2div(codeword, key);
    return (remainder.find('1') == string::npos);
}
int main() {
    string data = "10011101";
    string key = "1001"; // x^3 + 1
    cout << "=== TASK 2 ===" << endl;
    cout << "Data: " << data << endl;
    cout << "Key: " << key << endl;
    string codeword = encodeData(data, key);
    cout << "Transmitted Codeword: " << codeword << endl;
    // Introduce error at 3rd bit
    string received = codeword;
    received[2] = (received[2] == '0') ? '1' : '0';
    cout << "Received (error at bit 3): " << received << endl;

    if (receiverCheck(received, key))
        cout << "Receiver: No Error Detected" << endl;
    else
        cout << "Receiver: Error Detected" << endl;
    return 0;
}
```

Output: Finished | Clear Console

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