Bài tập tuần 1: Many Time Pad

1. Ý tưởng

- Vì 11 bản rõ đều được mã hoá bởi 1 key, nên ta có:
  Cypher[i] XOR CypherTarget
  = (Plain[i] XOR Key) XOR (PlainTarget XOR Key)
  = Plain[i] XOR PlainTarget
  Như vậy, kết quả của phép XOR 2 bản mã chính là kết quả của việc XOR 2 bản rõ.
- Khi thực hiện XOR 2 ký tự, nếu 1 ký tự là dấu cách, ký tự còn lại là 1 chữ cái thì kết quả của phép XOR sẽ chính là chữ cái đó bị đảo ngược hoa/thường.
- Bằng 2 ý tưởng trên, ta có thể đoán được ký tự của 2 bản rõ thông qua kết quả của phép XOR 2 bản mã.

2. Các bước thực hiện

- Lần lượt lấy 2 bit của bản mã cần giải mã XOR với 2 bit tương ứng của các bản mã còn lại.
- Kiểm tra kết quả phép XOR có cho ra chữ cái không?

```
48    for (let i = 0; i < input[10].length; i += 2) {
49        let isDecoded = false;
50        for (let j = 0; j < 10; j++) {
51            let s1 = '0x' + input[j][i] + input[j][i + 1];
52            let s2 = '0x' + input[10][i] + input[10][i + 1];
53            let strXOR = String.fromCharCode(s1 ^ s2 ^ 0x20)
54
55            // Kiểm tra xem ký tự vủa XOR có phải là chữ cái không
56            if (isCharacter(strXOR)) {
```

```
/**
 * Hàm kiểm tra xem ký tự có phải chữ cái không
 * @param str
 * @returns true nếu là chữ cái
 */
function isCharacter(str) {
    let patterm = /[a-z]/i;
    return patterm.test(str);
}
```

- Nếu kết quả cho ra chữ cái, khả năng cao phép XOR đó được thực hiện bởi 1 dấu cách và 1 chữ cái đó nhưng đảo ngược hoa thường. Ta cần kiểm tra xem ký tự đó của bản rõ là dấu cách hay chữ cái. Ta thực hiện XOR ký tự cần kiểm tra với các ký tự cùng thứ tự ở các bản mã còn lại. Trong 9 lần

kiểm tra, nếu có ít nhất 5 lần cho ra kết quả là 1 chữ cái thì ta thừa nhận rằng ký tự cần kiểm tra đó là dấu cách, ngược lại thì nó sẽ là chữ cái.

```
// Kiểm tra xem ký tự vừa XOR có phải là chữ cái không
if (isCharacter(strXOR)) {

    // Kiểm tra ký tự không ở bản cần giải mã có phải dấu cách không
    if (isSpace(i, j)) {
        // Nếu ký tự không ở bản cần giải mã là dấu cách thì ký tự cần giải mã chính là chữ cái
        targetPlain += strXOR;
        isDecoded = true;
        break;
    }
    if (isSpace(i, 10)) {
        // Nếu ký tự cần giải mã là dấu cách
        targetPlain += " ";
        isDecoded = true;
        break;
    }
}
```

```
/**
 * Hàm quyết định xem ký tự có phải là dấu cách không
 * @param i Ký tự thứ i cần kiểm tra
 * @param j Bản mã thứ j cần kiểm tra
 * @returns true nếu là dấu cách
 */
function isSpace(i, j) {
    let s1 = '0x' + input[j][i] + input[j][i + 1];
    let countIfCharacter = 0;
    for (let t = 0; t < 10; t++) {
        if (t != j) {
            let s2 = '0x' + input[t][i] + input[t][i + 1];
            if (s1 != s2) {
                let strXOR = String.fromCharCode(s1 ^ s2 ^ 0x20)
                if (isCharacter(strXOR)) countIfCharacter++;
            }
        }
    }
    return countIfCharacter >= 5;
}
```

- Thêm ký tự vừa giải mã được vào kết quả, sau đó tiếp tục việc giải mã các ký tự còn lại.

3. Mã nguồn
   https://github.com/chinhquoc01/Cryptography/blob/master/many-time-pad.js