



## **Laboratorio 2 - Parte 1: Algoritmos de detección Redes**

**Hugo Rivas - 22500**  
**Alexis Mesias - 22562**

Trama:

- Ingrese una trama en binario (ejemplo 110101): 110101110  
Mensaje original (bits): 110101110  
Trama Hamming generada : 0011101101110  
La trama Hamming ha sido guardada en 'emisor.txt'

Archivo donde se genera:

```
emisory.py  emisor.txt M X
emisor.txt
1 0011101101110
```

Bits cambiados de forma manual:

```
alexismesias@Alexiss-Mac-mini Lab2.1-Redes % g++ receptor.cpp -std=c++17 -o receptor
./receptor

Trama 1: 11100101101010011
- Sin errores. Mensaje original: 1010101010011

Trama 2: 00100100111010
- Sin errores. Mensaje original: 1010111010

Trama 3: 10100100111010011
- Sin errores. Mensaje original: 1010111010011

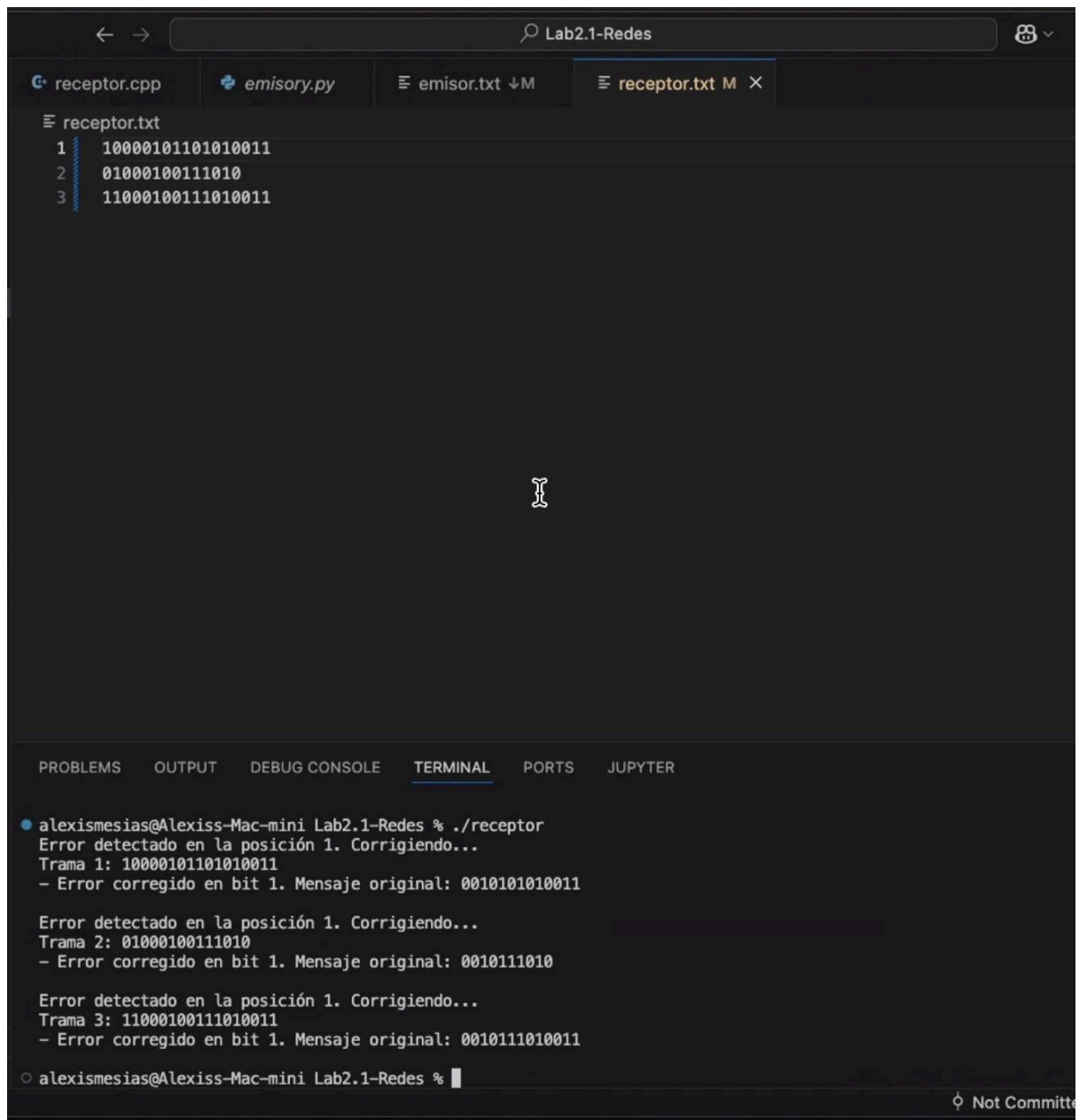
● alexismesias@Alexiss-Mac-mini Lab2.1-Redes % ./receptor
Error detectado en la posición 1. Corrigiendo...
Trama 1: 01100101101010011
- Error corregido en bit 1. Mensaje original: 1010101010011

Error detectado en la posición 1. Corrigiendo...
Trama 2: 10100100111010
- Error corregido en bit 1. Mensaje original: 1010111010

Error detectado en la posición 1. Corrigiendo...
Trama 3: 00100100111010011
- Error corregido en bit 1. Mensaje original: 1010111010011

○ alexismesias@Alexiss-Mac-mini Lab2.1-Redes %
```

Mensajes del receptor:



The image shows a Visual Studio Code editor window titled "Lab2.1-Redes". The editor has four tabs: "receptor.cpp", "emisory.py", "emisor.txt", and "receptor.txt". The "receptor.txt" tab is active, displaying the following content:

```
receptor.txt
1 10000101101010011
2 01000100111010
3 11000100111010011
```

Below the editor, the "TERMINAL" panel is open, showing the output of the command `./receptor` executed in the directory `Lab2.1-Redes`. The output shows three frames of data being processed, each with an error detected and corrected:

```
alexismesias@Alexiss-Mac-mini Lab2.1-Redes % ./receptor
Error detectado en la posición 1. Corrigiendo...
Trama 1: 10000101101010011
- Error corregido en bit 1. Mensaje original: 0010101010011

Error detectado en la posición 1. Corrigiendo...
Trama 2: 01000100111010
- Error corregido en bit 1. Mensaje original: 0010111010

Error detectado en la posición 1. Corrigiendo...
Trama 3: 11000100111010011
- Error corregido en bit 1. Mensaje original: 0010111010011

alexismesias@Alexiss-Mac-mini Lab2.1-Redes %
```

¿Es posible manipular los bits de tal forma que el algoritmo seleccionado no sea capaz de detectar el error? ¿Por qué sí o por qué no? En caso afirmativo, demuestrelo con su implementación.

En el caso del algoritmo de Hamming si es posible esconder un error si los bits a los que se le hace un flip son 3 o más flips que formen una palabra válida.

Link de github:

<https://github.com/Riv2oo4/Lab2.1-Redes.git>