*Marco Ramirez 21032 | Josué Morales 21116*

*[Company Name] | [Company Address]*

*Laboratorio No. 2*

Esquemas de detección y corrección: Parte 1 y 2

Tabla de contenidos

[Corrección de errores 2](#_Toc172730094)

[Algoritmo de Hamming 2](#_Toc172730095)

[Pruebas 2](#_Toc172730096)

[Detección de errores 4](#_Toc172730097)

[Fletcher checksum 4](#_Toc172730098)

[Pruebas 4](#_Toc172730099)

[Discusión 6](#_Toc172730100)

[Conclusiones 7](#_Toc172730101)

[Bibliografía 8](#_Toc172730102)

# Parte 1

## Corrección de errores

### Algoritmo de Hamming

Es una técnica de corrección de errores diseñada para asegurar la exactitud de los datos durante su transmisión o almacenamiento. Este código identifica y corrige errores que pueden surgir en el proceso de enviar o almacenar información desde el remitente hasta el receptor. (GeeksforGeeks, 2024)

#### Pruebas

(GeekforGeeks, Hamming code Implementation in Java , 2020)

* 0101011



* 000110



* 1000111

A screenshot of a computer error

Description automatically generated

## Detección de errores

### Fletcher checksum

Es un método de detección de errores utilizado por los protocolos de capa superior y se considera más fiable que el LRC, el VRC y el CRC. Este método emplea un Generador de Checksum en el lado del emisor y un Verificador de Checksum en el lado del receptor.

En el lado del emisor, el generador de checksum divide los datos en subunidades de n bits (generalmente de 16 bits), las suma usando el método de complemento a uno, y luego complementa el resultado. Este checksum complementado se añade al final de los datos originales y se envía al receptor. (GeekforGeeks, 2024)

#### Pruebas

(yadav, 2020)

* 1010001

A screenshot of a computer program

Description automatically generated

* 1101001

A screenshot of a computer program

Description automatically generated

* 0011101

A computer screen shot of a computer program

Description automatically generated

# Discusión

El análisis de los algoritmos de corrección y detección de errores, específicamente el código de Hamming y el checksum de Fletcher, tiene algunas características clave y limitaciones en la práctica.

Para el **algoritmo de Hamming** implementado en Java, observamos que la detección y corrección de errores están limitadas a un solo error por trama. Aunque el algoritmo es eficaz para corregir errores simples y detectar errores en una sola posición, no es capaz de identificar más de un error en la trama. Esto se debe a que el algoritmo se ajusta a una estructura específica que solo detecta el primer error. Una posible mejora sería la incorporación de recursividad en la función de corrección, lo que permitiría detectar y corregir múltiples errores. Sin embargo, se optó por mantener la estructura original para no desviarse de los estándares establecidos para el algoritmo.

Por otro lado, en el caso del **checksum de Fletcher** desarrollado en Python, se diseñó una función que generaba tramas de datos aleatorios para simular errores y verificar la capacidad del algoritmo para detectarlos. En los experimentos realizados con datos correctos, no se detectaron errores, pero el algoritmo mostró que podía identificar el primer error en las tramas alteradas. Similar al código de Hamming, el checksum de Fletcher también tiene la limitación de detectar solo el primer error.

# Conclusiones

* El código de Hamming y el checksum de Fletcher solo detectan y corrigen el primer error en una trama.
* Ambos métodos son eficaces para la corrección y detección de errores simples, pero limitados en escenarios con múltiples errores.
* El código de Hamming sigue estándares estrictos, lo que restringe su capacidad para detectar errores adicionales más allá del primero.
* El checksum de Fletcher, aunque flexible, también muestra limitaciones similares en la corrección de errores múltiples.

# Parte 2

# Bibliografía

GeekforGeeks. (11 de junio de 2020). *Hamming code Implementation in Java* . Obtenido de Geek for Geeks: https://www.geeksforgeeks.org/hamming-code-implementation-in-java/

GeekforGeeks. (7 de mayo de 2024). *Error Detection Code checksum*. Obtenido de Geek for geeks: https://www.geeksforgeeks.org/error-detection-code-checksum/

GeeksforGeeks. (17 de Junio de 2024). *Hamming Code in Computer Network*. Obtenido de Geek for Geeks: https://www.geeksforgeeks.org/hamming-code-in-computer-network/

yadav, C. (30 de junio de 2020). *Fletcher’s Checksum* . Obtenido de tutorial's point: https://www.tutorialspoint.com/fletcher-s-checksum