

## Taller 3

1. Debido a su capacidad para representar de manera más compacta y legible los datos binarios.

2. La longitud en términos de dígitos hexadecimales es un cuarto de la longitud en términos de dígitos binarios, y viceversa.

3. El sistema octal es un sistema numérico basado en la base 8, lo que significa que utiliza 8 dígitos: 0, 1, 2, 3, 4, 5, 6 y 7. Cada dígito octal representa 3 bits en binario.

- Relación con el binario:

• En el sistema octal, cada dígito puede ser representado por una secuencia de 3 bits en binario. Por ejemplo, el dígito octal 7 se representa en binario como 111.

• Para convertir un número binario a octal, se agrupan los dígitos binarios en bloques de 3 bits (de derecha a izquierda) y se convierten a su equivalente en octal.

- Relación con el hexadecimal:

• El sistema octal y el hexadecimal son menos directamente comparables que el octal y el binario, pero ambos sistemas son útiles en la representación de datos binarios.

• Cada dígito hexadecimal representa 4 bits, mientras que cada dígito octal representa 3 bits. Por lo tanto, dos dígitos octales equivalen a 6 bits, mientras que un dígito hexadecimal representa 4 bits.

4. El uso del sistema hexadecimal en programación ofrece varias ventajas frente al sistema decimal:

1. Compactación de datos binarios.

2. Facilidad de conversión.

3. Direcciones de memoria y códigos.

4. Legibilidad.

5. Compatibilidad con representaciones de color.

6. Facilidad en la depuración.



## Taller 3

5. El código Gray es un sistema de codificación binaria en el que dos números consecutivos difieren en solo un bit. El código Gray minimiza los errores al cambiar solo un bit a la vez.

Relación con los números binarios

1. Conversión de Binario a Gray:

- Para convertir un número binario a código Gray, el bit más significativo (MSB) del código Gray es el mismo que el MSB del número binario.

- Los bits restantes del código Gray se calculan utilizando la operación XOR, entre el bit actual del número binario y el bit anterior.

6.

1. Codificación de caracteres:

- ASCII asigna un valor numérico único a cada carácter imprimible y no imprimible (como caracteres de control).

- Los valores de ASCII van del 0 al 127, y cada uno de estos valores se representa en binario con 7 bits.

2. Representación Binaria:

- Cada carácter en ASCII se representa por una secuencia de 7 bits binarios.

- Por ejemplo, la letra mayúscula "A" en ASCII tiene el valor decimal 65, que se representa en binario como 01000001.

7.

Los números hexadecimales se utilizan para representar direcciones de memoria de manera compacta y legible.

Estas direcciones se convierten a binario para ser utilizadas en operaciones de bajo nivel, y el formato hexadecimal facilita la programación y depuración al proporcionar una representación más manejable de las ubicaciones en memoria.



## Taller 3

8.

Es mas eficiente por varias razones, especialmente en el contexto de Programacion y diseño de sistemas:

1. Compactacion de la representacion.
2. Facilidad de conversión
3. Legibilidad y manejo
4. Compatibilidad con Arquitectura del sistema
5. Facilitación de Operaciones Matemáticas
6. Uso en herramientas y depuración

9.

Los números binarios son preferidos para la logica de circuitos digitales por varias razones clave:

1. Simplicidad de implementación
2. Fiabilidad y Reducción de errores.
3. Estabilidad de señales
4. Eficiencia en circuitos Digitales
5. Facilidad de Construcción y Mantenimiento
6. Escalabilidad.

10.

Algunas formas en que se utilizan:

1. Direcciones de Memoria
2. Configuración de registros
3. Valores de datos
4. Depuración
5. Códigos de instrucción
6. Literales en el código.

11.

El sistema binario juega un papel fundamental en la transmisión de datos a través de redes informáticas.

1. Representación de datos
2. Codificación y Decodificación
3. Transmisión Física
4. Protocolos de comunicación



## Taller 3

### 5. Errores y corrección

12.

Más bits: mayor rango de valores que se pueden almacenar  
Datos sin signo: Tienen un rango positivo más amplio  
Datos con signo: Dividen el rango entre valores positivos y negativos.

El número de bits de cada tipo de dato, por tanto, influye directamente en el rango de números que puede representar

13.

- La ALU es el componente que realiza las operaciones aritméticas y lógicas dentro del procesador
- Las operaciones se realizan en base binaria, que es la base fundamental de la representación de datos en la informática
- La ALU maneja tanto las operaciones como la actualización de las banderas de estado para informar al proceso sobre el resultado de las operaciones.

Este diseño permite al procesador ejecutar operaciones matemáticas y lógicas rápidamente, lo que es crucial para la ejecución eficiente de programas

14.

Se refiere al orden en que se almacenan los bytes de un dato multi-byte en la memoria. Hay dos tipos principales de endianness

1. Big Endian
2. Little Endian.

¿CÓMO afecta a la representación de números?

1. En binario: La representación binaria del número no cambia, pero el orden en que los bytes están almacenados sí cambia.
2. En hexadecimal: la forma en que ves el número en hexadecimal puede parecer diferente dependiendo del endianness de la máquina.





## Taller 3

E 15/10/21

15.

Formas en que el sistema binario esta relacionado con los algoritmos de compresion:

1. Representación Binaria de Datos
2. Codificación Entropía y Binaria
3. Transformaciones y Representaciones Binarias
4. Compresion Basada en Patrones
5. Compresion de Datos Binarios Puros

16.

Un numero binario de 8 bits tiene 8 posiciones, por lo tanto 11111111  
 $= 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$

y en hexadecimal agrupamos 2 de 4 bits 1111 1111, lo cual cada uno equivalen a F y el resultado seria FF

17.

Un nibble es una secuencia de 4 bits. Dado que un bit puede ser 0 o 1, un nibble puede representar 16 combinaciones diferentes, desde 0000 hasta 1111 en binario.

18.

Para el color rojo = 255, Verde = 100, Azul = 50, la codificación hexadecimal es #FF6432, cada par de digitos en hexadecimal representa un componente de color en el rango de 00 a FF (0 a 255 en decimal), proporcionando un aspecto total de 16,7 millones de colores posibles ( $256 \times 256 \times 256$ ).

19.

Direcciones Mac

1. Formato y estructura
2. Representación Hexadecimal
3. Conversion a Binario y Decimal

Usos en Hardware y Redes

1. identificación Unica.
2. Configuración y Diagnostico
3. Compatibilidad y Estándares





## Taller 3

E 10/11/17

20.

En IPv6 se utiliza el sistema hexadecimal para representar las direcciones IP, con cada bloque de 16 bits traducido a 4 dígitos hexadecimales, y se aplican varias reglas para abreviar las direcciones y hacerlas más compactas.

21.

• Representación de todo tipo de datos: Los números binarios se utilizan para representar no solo números, sino también letras, imágenes, sonidos y cualquier otro tipo de información digital.

• Conversión a señales eléctricas: En los dispositivos electrónicos, los bits se representan como señales eléctricas: un voltaje alto para el 1 y un voltaje bajo para el 0.

22.

Muestreo: La señal de audio analógica (ondas sonoras) se convierte en una señal digital a través de un proceso llamado muestreo.

Cuantización: Cada muestra analógica se aproxima al valor digital más cercano. Este proceso se llama cuantización y es donde entra en juego el sistema binario.

• Codificación binaria: los valores cuantizados se representan como números binarios.

• Compresión (en el caso de MP3): Formatos como MP3 utilizan algoritmos de compresión para reducir el tamaño del archivo sin perder demasiada calidad de audio.

23.

Es una estrategia utilizada en sistemas digitales para manejar los desbordamientos que pueden ocurrir durante operaciones aritméticas. Se implementa en hardware a nivel de los circuitos aritméticos, como los sumadores y restadores.





## Taller 3

Ejemplo

24.

En graficas por computadora, las imagenes se representan como matrices de Pixeles. Cada Pixel tiene un valor que corresponde a su color. Este valor, a su vez, esta representado en binario.

25.

- Identificación Unica.
- Categorización
- Depuración
- Logs.
- Mensaje de error

1. Convierte 1000 en decimal, a binario y hexadecimal.

$$1000/2 = 500, \text{ resto } 0$$

$$500/2 = 250, \text{ resto } 0$$

$$250/2 = 125, \text{ resto } 0$$

$$125/2 = 62, \text{ resto } 1$$

$$62/2 = 31, \text{ resto } 0$$

$$31/2 = 15, \text{ resto } 1$$

$$15/2 = 7, \text{ resto } 1$$

$$7/2 = 3, \text{ resto } 1$$

$$3/2 = 1, \text{ resto } 1$$

$$1/2 = 0, \text{ resto } 1$$

$$= 1111101000$$

$$1000/16 = 62, 8$$

$$62/16 = 3, 14 = 3E8$$

$$3/16 = 0, 3$$

2. Convierte 3451 decimal, a binario y hexadecimal.

$$3451/2 = 1725, \text{ resto } 1$$

$$1725/2 = 862, \text{ resto } 1$$

$$862/2 = 431, \text{ resto } 0$$

$$431/2 = 215, \text{ resto } 1$$

$$215/2 = 107, \text{ resto } 1$$

$$107/2 = 53, \text{ resto } 1$$

$$53/2 = 26, \text{ resto } 1$$

$$26/2 = 13, \text{ resto } 0$$

$$13/2 = 6, \text{ resto } 1$$

$$6/2 = 3, \text{ resto } 0$$

$$3/2 = 1, \text{ resto } 1$$

$$1/2 = 0, \text{ resto } 1$$

$$= 110101111011$$

$$3451/16 = 215, 11$$

$$215/16 = 13, 7 = D7B$$

$$13/16 = 0, 13$$



# Taller 3

3. Convierte 11110101011 en binario, a decimal y hexadecimal.

$$1 * 1 = 1$$

$$1 * 2 = 2$$

$$0 * 4 = 0$$

$$1 * 8 = 8$$

$$0 * 16 = 0$$

$$1 * 32 = 32$$

$$0 * 64 = 0$$

$$1 * 128 = 128$$

$$1 * 256 = 256$$

$$1 * 512 = 512$$

$$1 * 1024 = 1024$$

$$= 1963$$

4. Convierte 101011000011 en binario, a decimal y ~~binario~~ hexa

$$1 * 1 = 1$$

$$1 * 2 = 2$$

$$0 * 4 = 0$$

$$0 * 8 = 0$$

$$0 * 16 = 0$$

$$0 * 32 = 0$$

$$1 * 64 = 64$$

$$1 * 128 = 128$$

$$0 * 256 = 0$$

$$1 * 512 = 512$$

$$0 * 1024 = 0$$

$$1 * 2048 = 2048$$

$$= 2755$$



### Taller 3

5. Convierte AFF en hexadecimal, a decimal y binario

1010 1111 1111

A F F  $\therefore 101011111111$

$$F \times 16^0 = 15 \times 1 = 15$$

$$F \times 16^1 = 15 \times 16 = 240$$

$$A \times 16^2 = 10 \times 256 = 2560$$

$$\left. \begin{array}{l} 15 \\ 240 \\ 2560 \end{array} \right\} = 2,815$$

6. Convierte D1B en hexadecimal, a decimal y binario

1101 0001 1011

D 1 B  $\therefore 110100011011$

$$B \times 16^0 = 11 \times 1 = 11$$

$$1 \times 16^1 = 1 \times 16 = 16$$

$$D \times 16^2 = 13 \times 256 = 3328$$

$$\left. \begin{array}{l} 11 \\ 16 \\ 3328 \end{array} \right\} = 3,355$$