

NAME  
Isaac Cabral

PAGES  
1/7

SPEAKER/CLASS  
RM

DATE - TIME  
02-10-2025

Title: Sistemas de Numeración

Keyword

Sistemas  
Numeros  
aditivos  
posicionales  
historia

Topic: Introducción

Notes:

Sistema aditivo: Se suman los valores de símbolos, sin importar la posición.

$1 = I$  ?  $10 = X$   $100 = C$  Sistema posicional  
el valor depende de la posición y lo hay

Egiptos Símbolos aditivos

Romanos: I, V, X, IV etc.

Babilonios: base 60, sin representación clara del cero.

Mayas: base 20, introducen el cero

Questions

¿Qué diferencia fundamental existe entre un sistema aditivo y uno posicional?

Summary:

Las primeras sistemas de numeración eran aditivos (egipcios, romanos), poco prácticos para cantidades grandes. Los babilonios desarrollaron un sistema posicional sexagesimal. Los mayas aportaron el uso del cero en un sistema vigesimal. Hoy se usan sistemas posicionales como el decimal, binario, octal etc.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Isaac Cabral	2/7	P.M	02-10-2025

Title: Sistemas Numericos

<p><b>Keyword</b></p> <p>Sistema decimal base 10 Notación posicional</p>	<p><b>Topic:</b> <u>Sistema Decimal</u></p> <p><b>Notes:</b></p> <p>Dígitos válidos: 0-9</p> <p>Representación Exponencial: cada cifra multiplicada por <math>10^n</math> según su posición.</p> <p>Ej: <math>836.74 = (8 \times 10^2) + (3 \times 10^1) + (6 \times 10^0) + (7 \times 10^{-1}) + (4 \times 10^{-2})</math>.</p>
<p><b>Questions</b></p>	<p>Es la referencia para convertir a otros sistemas</p> <p><math>1 = 10^0 \rightarrow</math> uno</p> <p><math>10 = 10^1 \rightarrow</math> diez</p> <p><math>100 = 10^2 \rightarrow</math> cien</p> <p><math>1.000 = 10^3 \rightarrow</math> mil</p> <p><math>10.000 = 10^4 \rightarrow</math> diez mil</p>

**Summary:** Sistema de base 10, el más usado, se basa en 10 dígitos (0-9). Utiliza notación posicional; el valor de cada cifra depende de su posición multiplicado por potencias de 10.



Title: Sistemas Numericos

Keyword

conversiones  
Bits  
Base 2  
Base 8  
Base 16

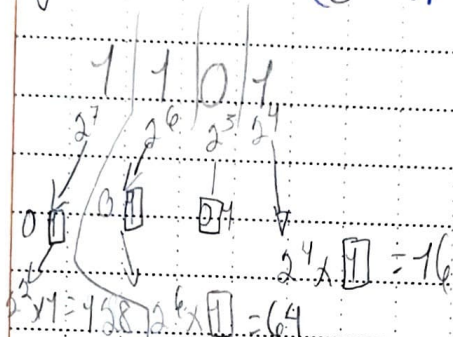
Topic:

Sistemas binario, octal y hexadecimal

Notes:

Binario: Fundamental en la computacion

Ej:  $10011(012) = 19.25(10)$



Questions

¿por que las computadoras usan el sistema binario?

¿como se convierten estos numeros binarios a octales o hexadecimales?

Octal simplifica binario (agrupa 3 bits)

Hexadecimal: usa las letras (A-F)

Binario	Octal	Hexadecimal
00000	0	0
00011	3	3
01010	A	12
01010	B	13
01111	F	17

Summary:

Sistemas posicionales con diferentes bases utilizadas en la informatica y la electrónica.  
Sistema binario (base 2): dígitos (0 y 1).  
Octal (base 8): dígitos (0-7).  
Hexadecimal (base 16): dígitos (0-9, A-F).

Title: Sistemas numericos

Keyword  
Complementos  
Binario  
Numeros Negativos  
Operacion  
Computadores

Topic: Suma de los complementos en complemento a 2

Notes: Complemento a 1: invertir cada bit  
(0 → 1, 1 → 0)

Complemento a 2: complemento a 1 + 1

Reglas:

Si el resultado tiene un acarreo final se descarta.

El resultado está en binario con signo positivo o negativo.

Questions

porque

Ej: Calcular  $7 - 5$  usando complementos a 2 en binario de 4 bits.

$7 = 0111$ ,  $5 = 0101$

Complemento a 2 de 5:

11 11 11 11 a 1 de 0101 → 1010

Sumando 1 → 1011

Sumando  $7 + (-5)$ :

$0010 = 2$

$7 - 5 = 2$

calcula  
como el resultado  
empieza con 1 es  
negativo.

Summary:

El complemento a 2 es un método usado en computación para representar números negativos y realizar sumas y restas en binario. Se invierten los bits (Complemento a 1) y se suma 1 al resultado para obtener el complemento a 2.



Title: Sistemas de numeración

<b>Keyword</b> Operaciones Base, suma Resta Multiplicación División	<b>Topic:</b> Operaciones básicas
	<b>Notes:</b> Las operaciones básicas son universales para cualquier sistema posicional. Debe realizarse en la misma base. Suma/Resta: Aplican reglas de números. Importante según la base. Multiplicación y División: mismo proceso que en decimal, pero cambiando la base. <b>Ej:</b> Suma de binario $1011_2 + 1101_2$ $1 + 1 = 10$ en binario $1011_2 + 1101_2 = 11000_2 = 24_{10}$ Resta en octal $725_8 - 157_8$ $725_8 - 157_8 = 546_8$
<b>Questions</b> ¿Que condición debe cumplirse antes de operar con números de distintos bases?	Multiplicación en hexadecimal (base 16) $A_{16} \times 5_{16} = 32_{16}$ $A_{16} = 10_{10}$ $10_{10} \times 5_{10} = 50_{10}$ $50_{10} = 3 \text{ resto } 2 \rightarrow 3/16 = 0 \text{ resto } 3$ $50_{10} = 32_{16}$ División en binario $11010_2 \div 10_2 = 1101_2$ En decimal: $11010_2 = 26_{10}$ $26 \div 2 = 13$ $13_{10} = 1101_2$

**Summary:** Las operaciones de suma, resta, multiplicación y división pueden realizarse en cualquier sistema numérico. Las reglas son las mismas que en decimal, pero adaptadas a la base correspondiente. Es indispensable que los números estén en la misma base antes de operar.

Title: Sistemas Numericos

Keyword

Conversion  
Base  
Decimal  
Exponente  
parte entera  
parte fraccionaria

Topic: Generalización de las conversiones

Notes: Es posible crear sistemas con cualquier base ejemplo (base 7, base 8)

Conversion general:

Sistema  $x \rightarrow$  Decimal: notación Exp.

Decimal  $\rightarrow$  Sistema  $n$ : division y multiplicacion de fracciones

Questions

Que procedimi-  
ento se sigue  
para convertir  
de decimal a  
otras bases

Ej: Convertir  $CD057_{EC}$  a base 20

$CD057_{15} = 418CH_{JECG}$

$$CD057_{EC_{15}} = 12 \times 15^4 + 13 \times 15^3 + 5 \times 15^2 + 7 \times 15^1 + 14 \times 15^0$$

Ahora lo convertiremos en base 20:

parte entera	Resto	parte fraccionaria
$651457/20 = 32572$	17	$0.90125 \times 20 = 18.025$
$32572/20 = 1628$	12	$0.732 \times 20 = 14.64$ entero
$1628/20 = 81$	8	$0.64 \times 20 = 12.8$
$81/20 = 4$	1	$0.8 \times 20 = 16.0$
$4/20 = 0$	4	

Summary:

Las reglas de conversión se aplican al cualquier sistema posicional, incluso se lo tiene no es una de las mas comunes (decimal binario octal etc).  
para convertir un sistema cualquiera (base  $x$ ) a decimal se usa notacion exponencial para pasar de decimal a otro sistema se divide entre la base  $x$  y se multiplica la fraccion por  $x$



Title: Sistemas numéricos

Keyword

computación  
Binario  
octal  
codificación  
Memoria

Topic:

Aplicación de los sistemas numéricos.

Notes:

Binario (base 2) usado en hardware  
lógica digital procesador: 00111, 0101, 111

Octal (base 8) simplifica cadenas de bits

Hexadecimal (base 16) útil en programación,  
dirección de memoria y representación de  
datos.

Ej: Dirección de Memoria  
En hexadecimal

En diseño gráfico un procesador puede  
usar cada represent: referencia a una celda  
de memoria con:

#RRGGBB

donde cada par es un  
número hexadecimal de 0 a FF (0-255 en decimal)

#FF0000 = Rojo (255, 0, 0)

#00FF00 = Verde (0, 255, 0)

#0000FF = Azul (0, 0, 255)

0x3F4A

base 16. En

decimales ser:

$3F4A_{16} = (3 \times 16^3 +$

$15 \times 16^2 + 4 \times 16^1 +$

$10 \times 16^0) = 16202_{10}$

Questions

¿Qué ventajas  
tienen octal  
y hexadecimal  
en la represen-  
tación de datos?

Summary:

Los sistemas numéricos tienen gran  
aplicación en computación. El binario es la base de  
los circuitos digitales, mientras que octal y hexadecimal  
se usan como formas abreviadas para representa-  
r cadenas largas de bits. También aplica en codifi-  
cación almacenamiento de datos, etc.