

## Sistemas de numeración

Un sistema de numeración es un conjunto de símbolos y reglas, que nos permiten construir todos los números válidos para el sistema. Los sistemas de numeración se clasifican en: posicionales y no posicionales.

En los sistemas posicionales: un número depende no sólo del símbolo utilizado sino también de la posición que ocupa dentro del número.

## Sistemas posicionales

Si el sistema posicional tiene *base*  $b$ , quiere decir que hay sólo  $b$  símbolos distintos para formar todos los números del sistema.

## Ejemplos

- Base 10: hay 10 símbolos distintos 0...9
- Base 2: hay 2 símbolos distintos 0 y 1.
- Base 8: hay 8 símbolos distintos 0...7

Cualquier número escrito en base 2, sólo puede tener 1 (unos) y 0 (ceros). (No aparecen 2...9)

Cualquier número escrito en base 8, sólo puede tener símbolos 0...7. (No aparecen 8 y 9).

Veamos el siguiente ejemplo

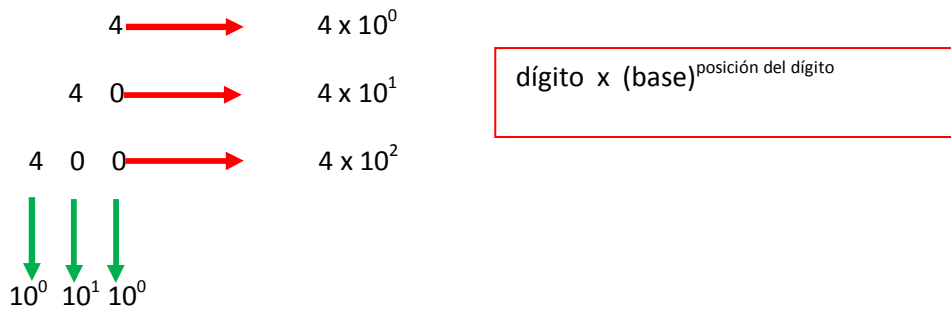
Supongamos trabajar en base 10.

- Escribamos el número cuatro: 4
- Escribamos el número cuarenta: 40
- Escribamos el número cuatrocientos: 400

Cada nuevo número es escrito con el mismo símbolo (4) desplazado a la izquierda ocupando una posición distinta:

- Unidad
- Decena
- Centena

Lo mismo podemos escribirlo de la siguiente manera:



En general podemos decir que en un sistema posicional cada número se genera usando los distintos dígitos de la base y cada uno de ellos resulta multiplicado por una potencia de la base, de acuerdo a la posición que ocupan.

Ejemplo

$$1895 = 1 \times 10^3 + 8 \times 10^2 + 9 \times 10^1 + 5 \times 10^0 = 1000 + 800 + 90 + 5 = 1895$$

Pensemos en el siguiente problema: dada una base y un cierto número de dígitos ¿cuántos números distintos podemos escribir?

Usemos un ejemplo para poder explicarlo:

En base 10:

- 1 dígito    0.....9     $\rightarrow$      $10^1=10$  números
- 2 dígitos    0...99     $\rightarrow$      $10^2=100$     “
- 3 dígitos    0...999     $\rightarrow$      $10^3=1000$     “

En general:                       $\text{cantidad de números distintos} = (\text{base})^{\text{cantidad de dígitos}}$



Recordar siempre. Para cualquier base.

Llegado a este punto vamos a analizar lo que hacemos (todos los días) para generar los distintos números en base 10 y luego lo extenderemos a otras bases.

Comenzamos con un dígito, desde el 0....9, el siguiente número se genera agregando un dígito a la izquierda 1 (es el que sigue al anterior) y repetimos la columna de la derecha de 0....9.

Cuando llegamos al final 19, aumentamos en 1 la columna de la izquierda y comenzamos otra vez, 20.....29.

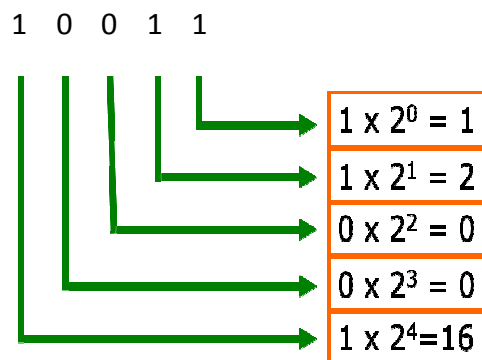
Cuando llegamos al último número (99), agregamos otra columna a la izquierda, repetimos el proceso generando el número 100 y así siguiendo. La siguiente tabla muestra lo explicado:

0	10	20	.....	90	100
1	11	21	.....	91	
2	12	22	.....	92	
3	13	23	.....	93	
4	14	24	.....	94	
5	15	25	.....	95	
6	16	26	.....	96	
7	17	27	.....	97	
8	18	28	.....	98	
9	19	29	.....	99	

Veamos estas ideas pero para base=2. Como mencionamos anteriormente hay sólo 2 símbolos distintos: 0 y 1. Cada uno de estos símbolos ó dígitos binarios se llama *bit*. Cualquier número estará formado por unos (1) y ceros (0) y no hay otro símbolo.

□ Números distintos =  $(2)^{\text{cantidad de dígitos}}$

Aquí las posiciones también son potencias de la base, pero ahora la base es 2. Veamos un ejemplo: dado el número  $10011_2$  ¿qué número en base 10 representa?



Hicimos lo mismo que en base 10, multiplicamos cada dígito (0,1) por una potencia de la base (2) que depende de la posición del dígito.

Finalmente el número equivalente en base 10 será:

$$16+0+0+2+1 = 19_{10}$$

$10011_2$  y  $19_{10}$  son números equivalentes escritos en distinta base.

El resultado recién alcanzado, se expresa en forma general:

$$N_{10} = \sum_{i=-m}^{i=n} d_i \times b^i = d_{-m} \times b^{-m} + d_{-m+1} \times b^{-m+1} + \dots + d_0 \times b^0 + d_1 \times b^1 + d_2 \times b^2 + \dots + d_n \times b^n$$

Se conoce como el *Teorema fundamental de la numeración*. N siempre quedará expresado en base 10, independientemente de la base b utilizada en el lado derecho de la igualdad.

Base mayor que 10: en este ejemplo base 16

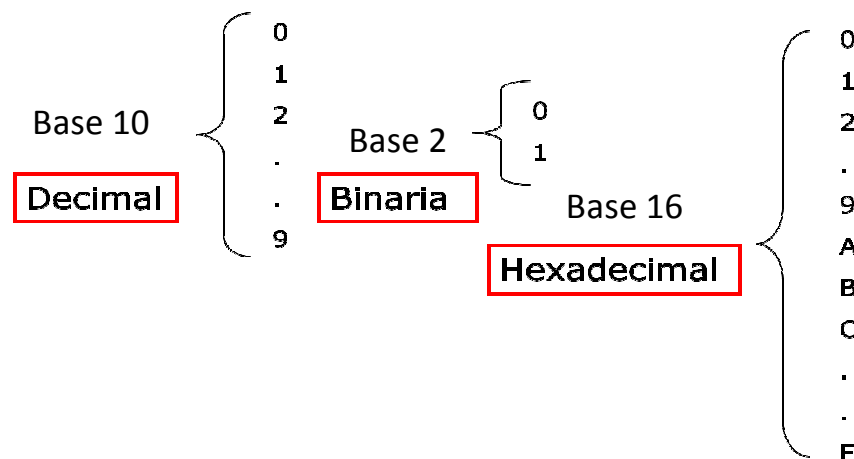
La base nos dice la cantidad de dígitos distintos que debe tener el sistema.

Si  $b=16$ , con los símbolos serán 0,1,2,...,9 hay 10 símbolos distintos. El símbolo que sigue al 9 no puede ser el 10 porque está formado por el 1 y 0 que ya fueron usados.

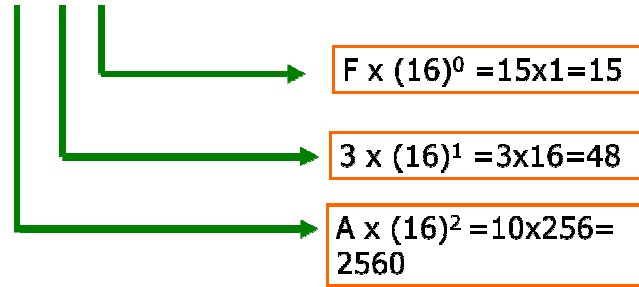
Entonces usamos letras A, B, C, D, E y F. Hasta la F porque ya tenemos los 16 símbolos distintos.

$A_{16}=10_{10}$   $B_{16}=11_{10}$  .....  $F_{16}=15_{10}$ .

Resumiendo



Veamos un ejemplo: A 3 F



Hicimos lo mismo que en base 10, multiplicamos cada dígito (0,1,2,...F) por una potencia de la base (16) que depende de la posición del dígito.

Finalmente el número equivalente en base 10 será:  $15 + 48 + 2560 = 2623_{10}$

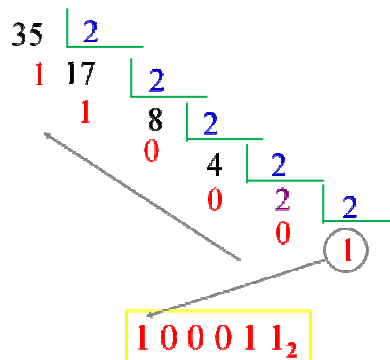
$A3F_{16}$  y  $2623_{10}$  son números equivalentes escritos en distinta base.

### Cambio de base

Para pasar un número de base 10 a cualquier otra base:

1. Dividir el número decimal por la nueva base. Guardar cociente y resto será el primer dígito del número en la nueva base.
2. Tomar el cociente anterior y repetir el paso 1 hasta que el cociente sea menor que la base y los distintos dígitos son los resultados de los distintos cocientes.
3. Escribir (concatenar) el último cociente y los restos empezando por el último.

Ej.  $35_{10} \longrightarrow ?_2$



En este ejemplo se muestra lo explicado anteriormente. El “último resto” de las divisiones sucesivas es el bit más significativo del resultado.

Otros dos ejemplos: cambiar de base los números 35 y 29 (base 10) a base 16.

Resultado  $\longrightarrow$

35	<u>16</u>
03	2
$23_{16}$	

29	<u>16</u>
13	1
$1D_{16}$	

En el segundo ejemplo vemos que el resto de la división es 13, pero este resto está escrito en base 10. El resultado debería estar expresado en base 16, por lo tanto el número 13 es reemplazado por la "letra" D que es el símbolo que representa al "13" en base 16.

Si el resto es 10 se reemplaza por A, si es 11 por B y así hasta 15 por F. Recordemos que esto pasa cuando la base es mayor que 10.

Por el teorema fundamental de la numeración, podemos escribir:

$$23_{16} = 2 \times 16^1 + 3 \times 16^0 = 32 + 3 = 35_{10}$$

$$1D_{16} = 1 \times 16^1 + 13 \times 16^0 = 16 + 13 = 29_{10}$$

Volvemos a obtener los números en base 10.

#### De base 2 a base 16

Dado un número en base 2, deseamos pasarlo a base 16. Hay una forma de hacerlo relativamente fácil debido a la relación que existe entre estas dos bases  $16 = 2^4$ . Vemos que una base hay que elevarla a la 4 para obtener la otra, pensando en esto podemos decir que por cada dígito hexadecimal hay 4 binarios. Siguiendo en la misma dirección si pensamos en la relación que hay entre la base 8 y la 2, sería  $8 = 2^3$ . Así por cada dígito octal hay 3 dígitos binarios por la relación que hay entre las dos bases.

Ej.

1	1	1	0	1	0	0	1	1	0	$= 3A6_{16}$
└───┘			└───┘			└───┘				
3			A			6				

#### De base 16 a base 2

Dado un número en base 16, para pasarlo a base 2 debemos reemplazar cada dígito hexadecimal por los 4 bits de su número binario equivalente.

	2	C	3	B
0010	1100	0011	1011	

*(Note: Blue arrows point from the hex digits 2, C, 3, B to their respective 4-bit binary representations above them.)*

Organización de Computadoras - Curso 2016  
Prof. Jorge Runco

DECIMAL	BINARIO	HEXADECIMAL	OCTAL
00	0000	0	00
01	0001	1	01
02	0010	2	02
03	0011	3	03
04	0100	4	04
05	0101	5	05
06	0110	6	06
07	0111	7	07
08	1000	8	10
09	1001	9	11
10	1010	A	12
11	1011	B	13
12	1100	C	14
13	1101	D	15
14	1110	E	16
15	1111	F	17