



# Computación 1

## Sistemas de Numeración

Curso 2021

Ingeniería Forestal

Universidad de la República

# Sistemas de Numeración

## Definición

- ⌚ Conjunto de símbolos y reglas sobre ellos, que se utilizan para la representación de cantidades.
- ⌚ La representación de una cantidad se efectúa mediante cadenas de símbolos.

# Sistemas de Numeración

## Clasificación

### No Posicionales

- ← El significado de cada símbolo no depende del lugar que ocupa en la cadena.

### Posicionales

- ← El significado de cada símbolo varía en función de la posición que ocupa en la cadena.

# Sistemas de Numeración

## Clasificación

### ⌚ Sistemas No Posicionales

- ← Marcas de conteo (I,II,III,IIII,IIII,...)
- ← Los porotitos y chapitas en el truco
  - ⌚ Cada poroto vale 1
  - ⌚ Cada chapita doblada vale X (3, 5 ...)
- ← Números romanos (subaditivo).
  - ⌚ La cadena XXXIII equivale al valor 33.
  - ⌚ El símbolo X aparece tres veces en la cadena.
  - ⌚ Siempre (casi) mantiene su valor: 10 unidades, sin importar el lugar que ocupa en la cadena.

# Sistemas de Numeración

## Clasificación

### Sistemas Posicionales

#### Utilizan un número Base

- Igual a la cantidad de símbolos que se utilizan para la representación.

 La cantidad representada se obtiene sumando el producto del valor de cada símbolo con la base elevada a la potencia correspondiente a la posición de dicho símbolo dentro de la cadena.

# Sistemas de Numeración

## Clasificación

### ⌚ Sistemas Posicionales

#### ⬅ Sistema decimal o de base 10

⌚ El que todos conocemos.

#### ⬅ Sistema binario o de base 2

⌚ Utilizado por los computadores para representar la información y con el que es capaz de trabajar

# Sistemas de Numeración

## Sistema Decimal

### Base 10

- ← Utiliza 10 símbolos diferentes (0, 1, 2, 3, 4, 5, 6, 7, 8 y 9)

### Explicación

- ← 10 dedos en las manos ?!

# Sistemas de Numeración

## Sistema Decimal

Las cifras se nombran de la forma:

$$N \equiv n_r \dots n_4 n_3 n_2 n_1 n_0 \quad n_i \in (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)$$

Representan el número en base 10:

$$N = \sum_{i=0}^r n_i * 10^i = n_r * 10^r + \dots + n_4 * 10^4 + n_3 * 10^3 + n_2 * 10^2 + n_1 * 10^1 + n_0 * 10^0$$

Ejemplo

$$\leftarrow 1357 = 1 \times 10^3 + 3 \times 10^2 + 5 \times 10^1 + 7 \times 10^0$$

# Sistemas de Numeración

## Sistema Binario

⌚ Sistema habitual en los sistemas electrónicos digitales.

⌚ Base 2

← Utiliza únicamente dos símbolos: 0 y 1.

# Sistemas de Numeración

## Sistema Binario

### 👤 Por qué?

- ⬅ Los computadores operan con componentes electrónicos que se encuentran "encendidos" o "apagados", correspondientes a 1 o 0.
  - sólo pueden procesar datos representados como tiras de ceros y unos.
- ⬅ Para poder interpretar la información que recibe del usuario el computador debe convertirla a binario.

# Sistemas de Numeración

## Sistema Binario

Los números se escriben de la siguiente forma:

$$B \equiv b_r \dots b_4 b_3 b_2 b_1 b_0 \quad b_i \in (0,1)$$

Representan la cantidad (en base 10):

$$B = \sum_{i=0}^r b_i * 2^i = b_r * 2^r + \dots + b_4 * 2^4 + b_3 * 2^3 + b_2 * 2^2 + b_1 * 2^1 + b_0 * 2^0$$

# Sistemas de Numeración

## Sistema Binario

👤 Valores de las posiciones.

← Primera  $2^0 = 1$

← Segunda  $2^1 = 2$

← Tercera  $2^2 = 4$

← .....

👤 Ejemplo

$$\leftarrow 1010_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 10_{10}$$

# Sistemas de Numeración

## Sistema Octal

### 👤 Base 8

↩ Utiliza 8 símbolos diferentes (0,1,2,3,4,5,6,7)

👤 Las cifras se nombran de la forma

$$O \equiv o_r \dots o_4 o_3 o_2 o_1 o_0 \quad o_i \in (0,1,2,3,4,5,6,7)$$

👤 Representa (en base 10):

$$O = \sum_0^r o_i * 8^i = o_r * 8^r + \dots + o_4 * 8^4 + o_3 * 8^3 + o_2 * 8^2 + o_1 * 8^1 + o_0 * 8^0$$

# Sistemas de Numeración

## Sistema Hexadecimal

### 👤 Base 16

- ← Utiliza 16 símbolos diferentes (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F).
  - 👤 Los diez primeros son los símbolos decimales y tienen el mismo significado que en la numeración decimal.
  - 👤 Los seis últimos son letras que representan: A=10, B=11, C=12, D=13, E=14 y F=15.

# Sistemas de Numeración

## Sistema Hexadecimal

Las cifras se nombran de la forma:

$$H \equiv h_r \dots h_4 h_3 h_2 h_1 h_0 \quad h_i \in (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)$$

Representa (en base 10):

$$H = \sum_{i=0}^r h_i * 16^i = h_r * 16^r + \dots + h_4 * 16^4 + h_3 * 16^3 + h_2 * 16^2 + h_1 * 16^1 + h_0 * 16^0$$

Ejemplo

$$\leftarrow E07F_{16} = 14 \times 16^3 + 0 \times 16^2 + 7 \times 16^1 + 15 * 16^0 = 57471_{10}$$

# Sistemas de Numeración

## Cambio de base

- 👤 De base  $b$  a base 10 (otra forma de verlo):
- ← Inicializar el resultado con la cifra más significativa
  - ← Hasta que no queden cifras:
    - Multiplicar el resultado por  $b$  y sumarle la siguiente cifra

# Sistemas de Numeración

## Cambio de base

👤 Ejemplo:

$102012_3$  (base 3)

» res = **1**      %primera cifra

» res = res\*3+**0** = 3

» res = res\*3+**2** = 11

» res = res\*3+**0** = 33

» res = res\*3+**1** = 100

» res = res\*3+**2** = 302

# Sistemas de Numeración

## Cambio de base

### De base 10 a base $b$

- ← Se divide el número entre  $b$  y se toma el resto.
- ← Se vuelve a dividir el cociente obtenido en la división anterior entre  $b$  y se toma nuevamente el resto.
- ← Esta operación se repite hasta que el cociente resultante sea menor que  $b$ .
- ← Por último se escribe, en este orden, el último cociente, el último resto, el penúltimo resto, el antepenúltimo resto,...



# Sistemas de Numeración

## Cambio de base

⌚ Método rápido de decimales a binarios.

1. Dado el número decimal identificar la mayor potencia de 2 contenida en el.
2. Colocar 1 en la posición correspondiente a dicha potencia y restar la potencia al número.
3. Preguntar si cabe la siguiente potencia de 2 en orden decreciente.

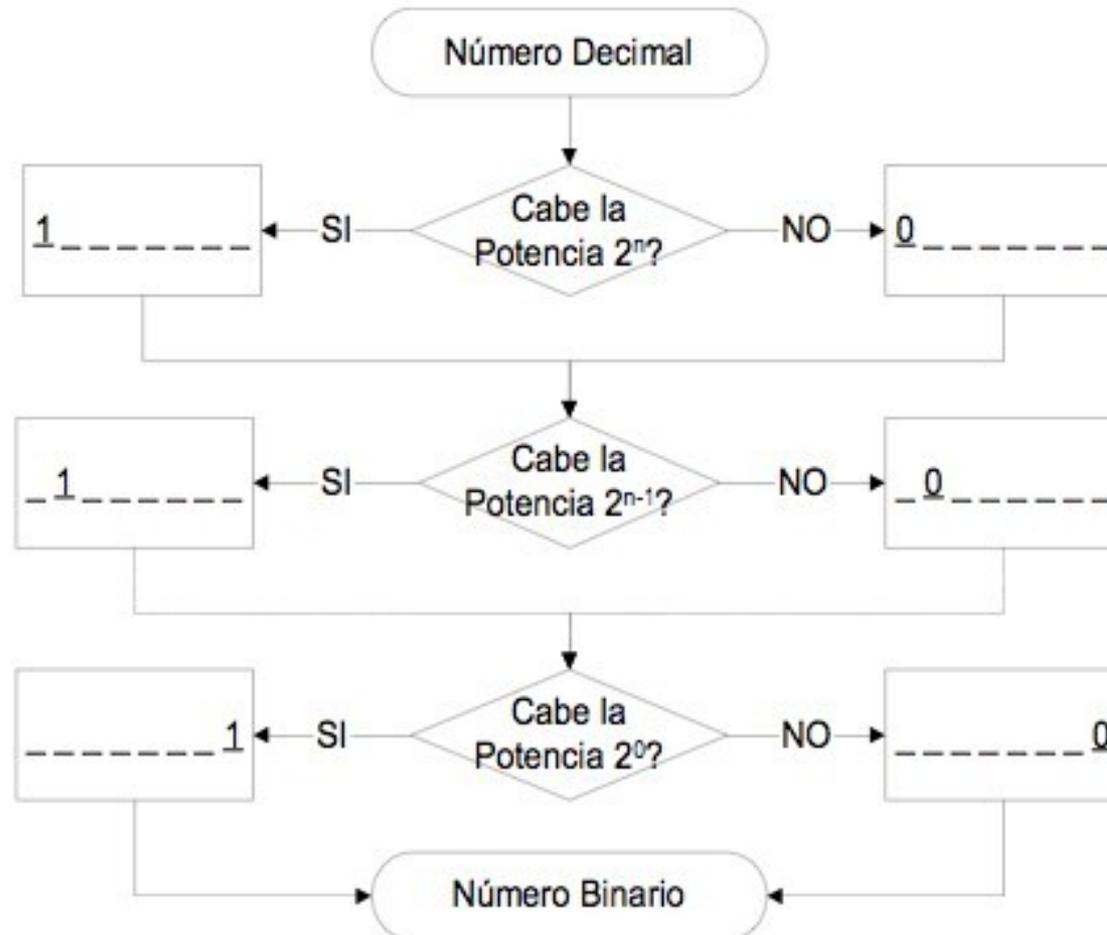
⌚ Si cabe coloco el símbolo 1 en la posición correspondiente a la potencia y resto la potencia al número.

⌚ Si no cabe coloco 0 en la posición correspondiente a la potencia.

4. Repetir el procedimiento hasta llegar a  $2^0$ .

# Sistemas de Numeración

## Cambio de base



# Sistemas de Numeración

## Cambio de base

Convertir 135

- ↩  $135 - 2^7 = 135 - 128 = 7$  entonces  $2^7$  cabe
- ↩  $7 - 2^6 = 7 - 64 = -57$  entonces  $2^6$  no cabe
- ↩  $7 - 2^5 = 7 - 32 = -25$  entonces  $2^5$  no cabe
- ↩  $7 - 2^4 = 7 - 16 = -9$  entonces  $2^4$  no cabe
- ↩  $7 - 2^3 = 7 - 8 = -1$  entonces  $2^3$  no cabe
- ↩  $7 - 2^2 = 7 - 4 = 3$  entonces  $2^2$  cabe
- ↩  $3 - 2^1 = 3 - 2 = 1$  entonces  $2^1$  cabe
- ↩  $1 - 2^0 = 1 - 1 = 0$  entonces  $2^0$  cabe

$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
1	0	0	0	0	1	1	1

# Sistemas de Numeración

## Cambio de base

### ⌚ Casos particulares bases 8 y 16.

← La base 8 (octal) y la base 16 (hexadecimal) tienen una íntima relación con la base 2.

← Octal a binario y viceversa

⌚  $8 = 2^3$

⌚ Cada símbolo octal corresponde a 3 símbolos binarios

← Hexadecimal a binario y viceversa

⌚  $16 = 2^4$

⌚ Cada símbolo hexa corresponde a 4 símbolos binarios

# Sistemas de Numeración

## Cambio de base

### Binario a Octal

- ← Dividir en grupos de 3 bits a partir del punto binario.
- ← Asignarle a cada grupo el símbolo octal correspondiente.

Convertir  $11001010011_2$  a base 8

$$\begin{array}{cccc} \underline{11} & \underline{001} & \underline{010} & \underline{011} \\ 3 & 1 & 2 & 3 \end{array} \quad = 3123_8$$

# Sistemas de Numeración

## Cambio de base

### ↻ Octal a Binario

- ↻ Inverso del caso anterior
- ↻ Convertir a binario cada símbolo octal
- ↻ Ejemplo convertir  $732_8$

$$\text{↻ } 7_8 = 111_2$$

$$\text{↻ } 3_8 = 011_2 \Rightarrow 732_8 = 111011010_2$$

$$\text{↻ } 2_8 = 010_2$$

# Sistemas de Numeración

## Cambio de base

### 👤 Binario a Hexadecimal

- ⬅ Dividir el número hexa en grupos de 4 bits.
- ⬅ Asignar a cada grupo el símbolo hexadecimal correspondiente.
- ⬅ Ejemplo:

$$\underline{1101} \ \underline{1011} \ \underline{1000} \ \underline{0110}_2 = \text{DB86}_{16}$$

D      B      8      6

# Sistemas de Numeración

## Cambio de base

### Hexadecimal a Binario

- ← Inverso del caso anterior.
- ← Convertir cada símbolo hexadecimal a binario.

# Sistemas de Numeración

## Equivalencias

Decimal	Binario	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9

Decimal	Binario	Octal	Hexadecimal
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13

# Sistemas de Numeración

## Ejercicio

Implementar una función que permita realizar el cambio de base de un decimal hacia otra base  $b$

### Entrada

← Número decimal

← Base  $b$

### Salida

← Vector conteniendo la representación del decimal en la base  $b$

# Sistemas de Numeración

## Ejercicio

```
function y = CambioBase(numero, base)
resto = 0;
cociente = numero;
resultado = [];
while cociente >= base
    resto = mod(cociente, base);
    cociente = floor(cociente/base);
    resultado = [resto resultado];
end
y = [cociente resultado]
```

# Operaciones

⌚ Es exactamente igual a lo que ya conocemos !!!!

⌚ Se cumple todo lo que aprendimos en la escuela ! (pero mucho más fácil !)

# Operaciones

## 👤 Sumas

👤 Empecemos por algo desafiante ..

# Operaciones

## 👤 Sumas

👤 Empecemos por algo desafiante ..

$$\begin{array}{r} 1 \\ 1 \\ \hline \end{array}$$

# Operaciones

## Sumas

Empecemos por algo desafiante ..

$$\begin{array}{r} 1 \\ 1 \\ \hline 10 \end{array}$$

# Operaciones

## Sumas

← Algo un poco más complicado

$$\begin{array}{r} 1000110 \\ 10111011 \\ \hline \end{array}$$

# Operaciones

## Sumas

← Algo un poco más complicado

$$\begin{array}{r} 1000110 \\ 10111011 \\ \hline 100000001 \end{array}$$

# Operaciones

## Restas

 Se puede verificar ..

$$\begin{array}{r} 100000001 \\ 1000110 \\ \hline 10111011 \end{array}$$

# Operaciones

## ⌚ Multiplicación

⌚ El mismo algoritmo que ya conocemos.

$$\begin{array}{r} 10001 \\ \times 11 \\ \hline 10001 \\ 10001 \\ \hline 110011 \end{array}$$

# Operaciones

## ✎ Multiplicación

- ✎ Qué pasa al multiplicar por 2 en binario?
  - (en cualquier base al multiplicar por la base)
- ✎ Qué pasa al multiplicar un número en binario por otro (en binario) que es un 1 y varios 0s?
  - (la misma idea en otra base)
- ✎ Cuántas cifras tiene el resultado de la multiplicación de dos números binarios ?

# Operaciones

## ⌘ División

⌘ El mismo “algoritmo” que ya conocemos funciona, pero toda la operativa en binario !!!