

Que vimos?

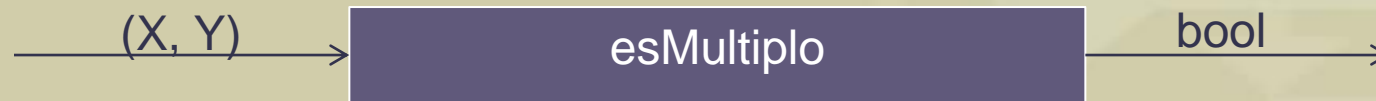
- Algoritmo: Procedimiento a seguir para resolver un problema definido en términos de:
 - Las acciones a ejecutar
 - El orden de dichas acciones
- Control de flujo:
 - Secuencial
 - Condicional
 - Repetitivo

Que vimos?

- Modela, divide y conquista
 - Modelado: identificación y caracterización de las entidades involucradas en un problema
 - Diseño: la identificación de subproblemas cumple un rol fundamental (tarea de diseño) y sienta las bases para la descomposición del problema.
 - Función: definen la abstracción a la solución de cada subproblema individual. Ladrillo constructivo.

Funciones

Funciones como cajas negras:




```
// función que retorna verdadero si el valor de X es múltiplo de Y,  
// o retorna falso en caso contrario
```

```
esMultiplo( X, Y )  
    si modulo(X, Y) = 0  
        retornar verdadero;  
    sino  
        retornar falso;  
    fin si  
fin esMultiplo
```

Importancia de una buena elección del algoritmo

tamaño	n^2	t [seg]		$n \cdot \log(n)$	t [seg]	
10	100	0.1		33	0.03	
100	10000	10		664	0.66	
1000	1000000	1000		9966	9.97	
1000000	1E+12	1E+09		19931569	19,931.57	
		11574.07	días		5.54	Horas
		31.70979	años			

Representación



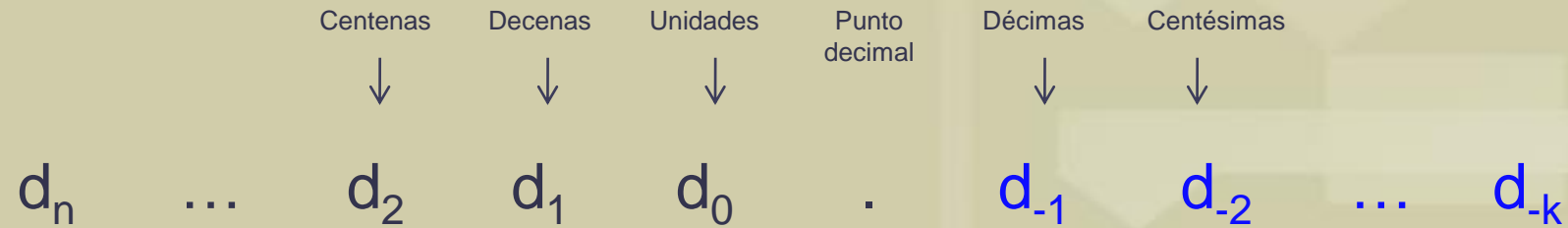
Que vimos?

- Números de precisión finita.
 - El orden de las operaciones pueden afectar el resultado final.
 - Se pierden algunas propiedades (cierre, distributiva, asociativa)
- La importancia de conocer que hay limitaciones y que hay que convivir con ellas.

Sistemas de numeración

Número decimal, en base 10:

2009.14



$$\text{Número} = \sum_{i=-k}^n d_i 10^i$$

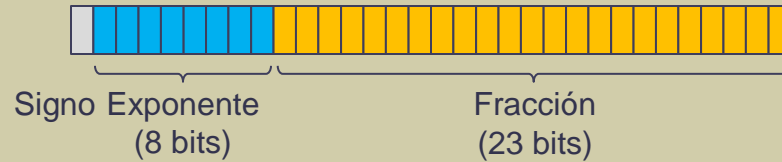
$$2009.14 = 2 \times 10^3 + 0 \times 10^2 + 0 \times 10^1 + 9 \times 10^0 + 1 \times 10^{-1} + 4 \times 10^{-2}$$

Como se representan los datos en la compu?

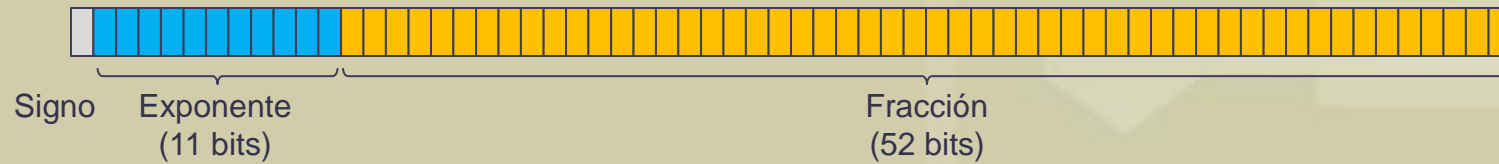
- Siempre en representaciones en base 2
- Números enteros positivos:
 - representación binaria del número.
- Números enteros con signo:
 - representación binaria para positivos.
 - Complemento a 2 para negativos
- Números con punto flotante:
 - IEEE 754 de 32 bits
 - IEEE 754 de 64 bits

IEEE 754

Simple precisión (32 bits) 1 bit signo, 8 bits exponente y 24 bits de fracción



Doble precisión (64 bits) 1 bit signo, 11 bits exponente y 53 bits de fracción



En la fracción se omite el primer bit que siempre es 1.

El exponente está codificado con un offset (excess) igual a $(2^{e-1})-1$.

$$\text{Número} = (-1)^{\text{Signo}} \times 2^{\text{Exponente}-\text{Offset}} \times (1 + \text{Fracción})$$