

AYUDANTÍA S. 3A

PÉRDIDA DE IMPORTANCIA



Ayudante: Francisco Manríquez Novoa



Jueves 27 de marzo del 2025



Computación Científica

Notación científica

Un número en notación científica se expresa como un **signo + o -** por el producto entre dos números:

- Un valor m ($1 \leq |m| < 10$), a cuya parte decimal se le dice “**mantisa**”
 - (aunque ese no es el significado original de “mantisa”)
- El orden de magnitud 10^n , donde n es el **exponente** del número

$$\text{signo } [-] 1,602 \times 10^{-19}$$

mantisa

exponente

Suma en base 10 [1]

¿Cómo sumamos 2 números en notación científica?

$$\begin{array}{r} 1.476 \times 10^6 \\ + 4.512 \times 10^3 \end{array}$$

Suma en base 10 [2]

¿Cómo sumamos 2 números en notación científica?

- El número de mayor exponente domina al otro
- El de menor exponente debe ajustarse

$$\begin{aligned} &1.476 \times 10^6 \\ &+ 4.512 \times 10^3 \end{aligned}$$

Suma en base 10 [3]

¿Cómo sumamos 2 números en notación científica?

- El número de mayor exponente domina al otro
- El de menor exponente debe ajustarse

$$\begin{array}{r} 1.476 \times 10^6 \\ + 4.512 \times 10^3 \end{array}$$

Suma en base 10 [4]

¿Cómo sumamos 2 números en notación científica?

- El número de mayor exponente domina al otro
- El de menor exponente debe ajustarse

$$\begin{array}{r} 1.476 \times 10^6 \\ + 0.004512 \times 10^6 \end{array}$$

Suma en base 10 [5]

¿Cómo sumamos 2 números en notación científica?

- El número de mayor exponente domina al otro
- El de menor exponente debe ajustarse

$$\begin{array}{r} 1.476 \times 10^6 \\ + 0.004512 \times 10^6 \\ \hline 1.480512 \times 10^6 \end{array}$$

Suma en base 10 [6]

Mini-ejercicio: sumar estos dos números:

$$\begin{array}{r} 3.14 \times 10^3 \\ + 2.71 \times 10^{-9} \end{array}$$

Suma en base 10 [7]

Mini-ejercicio: sumar estos dos números:

$$\begin{array}{r} 3.14 \times 10^3 \\ + 2.71 \times 10^{-9} \times 10^{12} \end{array}$$

Suma en base 10 [8]

Mini-ejercicio: sumar estos dos números:

$$\begin{array}{r} 3.14 \times 10^3 \\ + 0.0000000000000000271 \times 10^3 \end{array}$$

Suma en base 10 [9]

Mini-ejercicio: sumar estos dos números:

$$\begin{array}{r} 3.14 \times 10^3 \\ + 0.0000000000000000271 \times 10^3 \\ \hline 3.1400000000000000271 \times 10^3 \end{array}$$

Suma en base 10 [10]

Si la mantisa tiene demasiados números decimales, puede que debamos redondear. Depende de cuántos decimales podemos almacenar.

$$3.14000000000000271 \times 10^3 \\ \approx 3.14 \times 10^3$$

Redondeo en base 10 [1]

Si queremos redondear estos números a una sola cifra decimal, redondeamos al valor más cercano:

$$1.42 \approx 1.4 \quad \downarrow$$

$$1.47 \approx 1.5 \quad \uparrow$$

Redondeo en base 10 [2]

...pero, si nuestro número está justo “a la mitad”,
¿a qué valor redondeamos?

$$1.45 \approx \begin{matrix} & 1.4 \\ < & \\ & 1.5 \end{matrix}$$

Redondeo en base 10 [3]

Convención: “Si el dígito anterior es par, redondear hacia abajo. Si es impar, hacia arriba” (enseñado en FIS100).

$$1.4 < 1.\boxed{4}5 < 1.5$$
$$\Rightarrow 1.45 \approx 1.4 \downarrow$$

$$1.7 < 1.\boxed{7}5 < 1.8$$
$$\Rightarrow 1.75 \approx 1.8 \uparrow$$

Punto flotante en base 2

Usamos la misma idea: representar números binarios de una forma similar a la “notación científica”.

Esta vez, usamos una potencia de 2.

$$1.0010 \times 2^3$$

Suma en base 2 [1]

Para sumar dos números binarios, de nuevo: ajustamos el número de menor exponente para coincidirlo con el mayor.

$$\begin{array}{r} 1.01 \times 2^5 \\ + 1.11 \times 2^3 \end{array}$$

Suma en base 2 [2]

Para sumar dos números binarios, de nuevo: ajustamos el número de menor exponente para coincidirlo con el mayor.

$$\begin{array}{r} 1.01 \times 2^5 \\ + 1.11 \times 2^{-2} \times 2^3 \times 2^2 \end{array}$$

Suma en base 2 [3]

Para sumar dos números binarios, de nuevo: ajustamos el número de menor exponente para coincidirlo con el mayor.

$$\begin{array}{r} 1.01 \times 2^5 \\ + 0.0111 \times 2^5 \end{array}$$

Suma en base 2 [4]

Para sumar dos números binarios, de nuevo: ajustamos el número de menor exponente para coincidirlo con el mayor.

$$\begin{array}{r} 1.01 \times 2^5 \\ + 0.0111 \times 2^5 \\ \hline 1.1011 \times 2^5 \end{array}$$

Redondear punto flotante [1]

Por lo general, redondeamos al valor más cercano.

Ejemplo: redondear estos dos números a una cifra después del punto:

$$1.001 \approx 1.0 \quad \downarrow$$

$$1.011 \approx 1.1 \quad \uparrow$$

Redondear punto flotante [2]

¿Qué pasa si el número está justo al medio?

$$1.01 \approx \begin{matrix} \nearrow 1.0 \\ \searrow 1.1 \end{matrix}$$

Redondear punto flotante [3]

Convención: “Si el dígito anterior es par (0), redondear hacia abajo. Si es impar (1), hacia arriba” (enseñado en FIS100).

$$1.0 < 1.01 < 1.1$$
$$\Rightarrow 1.01 \approx 1.0 \quad \downarrow$$

$$1.1 < 1.\boxed{1}1 < 10.0$$
$$\Rightarrow 1.11 \approx 10.0 \quad \uparrow$$



Redondear punto flotante [4]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10111

Redondear punto flotante [5]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10111

$0.101 < 0.10111 < 0.110$

Redondear punto flotante [6]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10111

$0.101 < 0.10111 < 0.110$

Está más cerca del número de arriba que del de abajo, así que se redondea hacia arriba (0.110).



Redondear punto flotante [7]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

1.00110

Redondear punto flotante [8]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

1.00110

$1.001 < 1.00110 < 1.010$

Redondear punto flotante [9]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

1.00110

$1.001 < 1.00110 < 1.010$

Está justo a la mitad entre ambos números. Como el tercer decimal es 1 (impar), se redondea hacia arriba (1.010).



Redondear punto flotante [10]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10010

Redondear punto flotante [11]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10010

$0.100 < 0.10010 < 0.101$

Redondear punto flotante [12]

Ejercicios: redondear los siguientes números a 3 cifras después del punto.

0.10010

$0.100 < 0.10010 < 0.101$

Está justo a la mitad entre ambos números. Como el tercer decimal es 0 (par), se redondea hacia abajo (0.100).

Pérdida de importancia

Al sumar, por ejemplo, 1 con 2^{-100} , se obtiene el número binario 1.000...001 (100 bits después del punto).

Los 100 bits de la parte fraccionaria no caben en los 52 bits de la mantisa en double precision.

Por lo tanto, el resultado se redondea a 1. En este caso, ocurrió **pérdida de importancia**: 2^{-100} era muy poco importante con respecto a 1 y su contribución al resultado final se perdió con el redondeo.



En máquina [1]

La máquina puede operar (sumar, multiplicar...) números como si tuviera infinitos decimales y solo redondea el resultado FINAL de cada operación. **NO SE REDONDEA CADA TÉRMINO ANTES DE OPERAR.**

¿Cómo hace eso la máquina? Para más detalles:

https://en.wikipedia.org/wiki/Floating-point_arithmetic#Floating-point_operations

“...it would appear that a large number of extra digits would need to be provided by the adder to ensure correct rounding; however, for binary addition or subtraction using careful implementation techniques **only a guard bit, a rounding bit and one extra sticky bit** need to be carried beyond the precision of the operands.”



En máquina [2]

Consecuencias: podemos sumar y restar números muy diferentes en orden de magnitud. La máquina los maneja de manera exacta sin necesitar infinitos bits de precisión. Solo se redondea el resultado.

Por ejemplo, podemos sumar o restar 1 con 2^{-53} , cuya diferencia de orden de magnitud es 53.



En máquina [3]

SUMA: ¿cuánto es $1 + 2^{-53}$?

En máquina [4]

SUMA: ¿cuánto es $1 + 2^{-53}$?

$$\begin{array}{r} 1 \\ + 2^{-53} \end{array}$$

En máquina [4]

SUMA: ¿cuánto es $1 + 2^{-53}$?

$$\begin{array}{r} 1 \\ + 2^{-53} \end{array}$$

En máquina [5]

SUMA: ¿cuánto es $1 + 2^{-53}$?

$$\begin{array}{r} 1 \\ + 0. \underbrace{000 \dots 01}_{53 \text{ bits}} \end{array}$$

En máquina [6]

SUMA: ¿cuánto es $1 + 2^{-53}$?


$$\begin{array}{r} 1 \\ + 0.000\dots01 \\ \hline 1.000\dots01 \end{array}$$

$\underbrace{\hspace{1.5cm}}$
53 bits

En máquina [7]

SUMA: ¿cuánto es $1 + 2^{-53}$?

$$\begin{array}{r} 1 \\ + 0.000\dots01 \\ \hline 1.000\dots01 \end{array}$$


53 bits

La mantisa (53 bits)
NO cabe en double
precision (52 bits).

$1 + 2^{-53}$ no se puede
representar de
manera exacta. Se
debe redondear a 1.



En máquina [8]

RESTA: ¿cuánto es $1 - 2^{-53}$?

En máquina [9]

RESTA: ¿cuánto es $1 - 2^{-53}$?

$$1 - 2^{-53}$$

En máquina [10]

RESTA: ¿cuánto es $1 - 2^{-53}$?

$$\begin{array}{r} 1 \\ - 0.000\dots01 \\ \hline \end{array}$$

53 bits

En máquina [11]

RESTA: ¿cuánto es $1 - 2^{-53}$?

$$\begin{array}{r} 1 \\ -0.000\dots01 \\ \hline 0.\underbrace{111\dots11}_{53 \text{ bits}} \end{array}$$

En máquina [12]

RESTA: ¿cuánto es $1 - 2^{-53}$?

$$\begin{array}{r} 1 \\ -0.000\dots01 \\ \hline 0.\underbrace{111\dots11}_{53 \text{ bits}} \\ \hline \underbrace{1.11\dots11}_{52 \text{ bits}} \times 2^{-1} \end{array}$$

En máquina [13]

RESTA: ¿cuánto es $1 - 2^{-53}$?

$$\begin{array}{r} 1 \\ -0.000\dots01 \\ \hline 0.\underbrace{111\dots11}_{53 \text{ bits}} \\ \hline \underbrace{1.11\dots11}_{52 \text{ bits}} \times 2^{-1} \end{array}$$

La mantisa (52 bits)
Sí cabe en double
precision (52 bits).

$1 - 2^{-53}$ sí se puede
representar de
manera exacta con
mantisa 11...11 y
exponente -1.

Ejercicio 1

$$\cos(x) \approx 1 - \frac{x^2}{2}$$

Al ingresar una potencia de 2, $x = 2^{-n}$, ¿cuál es el mínimo valor de n tal que hay pérdida de importancia al evaluar esta aproximación de $\cos(2^{-n})$ en double precision? Usa todo lo aprendido hasta ahora: sumas, restas, contar bits...

Spoiler: no es $n = 26$, sino $n = 27$.

Ejercicio 2

$$f(x) = x - \frac{x^3}{4}$$

Al ingresar una potencia de 2, $x = 2^{-n}$, ¿cuál es el mínimo valor de n tal que hay pérdida de importancia al evaluar $f(2^{-n})$ en double precision? Usa todo lo aprendido hasta ahora: sumas, restas, contar bits...

Spoiler: cualquier $n > 25.5$ provoca pérdida de importancia. El mínimo es $n = 26$.



¿Dudas?