

GLI ERRORI



Problema

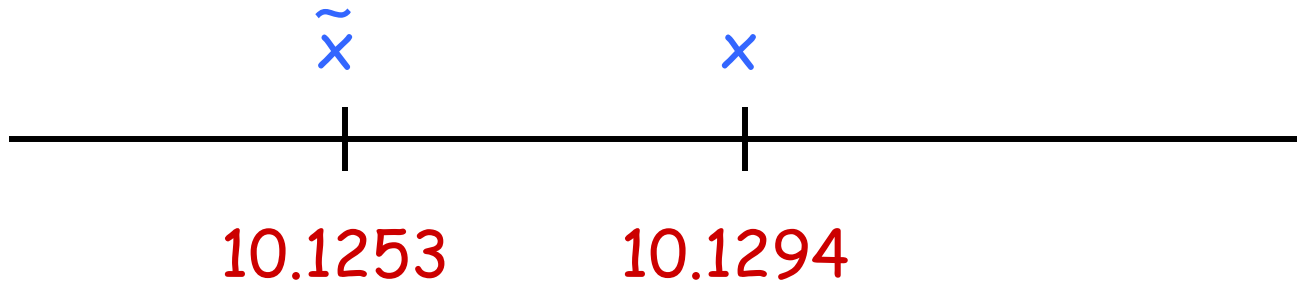
valutare l'**accuratezza**
del risultato di un calcolo e
quindi l'**affidabilità** del risultato stesso

ERRORE ASSOLUTO

Esempio

Si vuole approssimare

$$x = 10.1294 \quad \text{con} \quad \tilde{x} = 10.1253$$



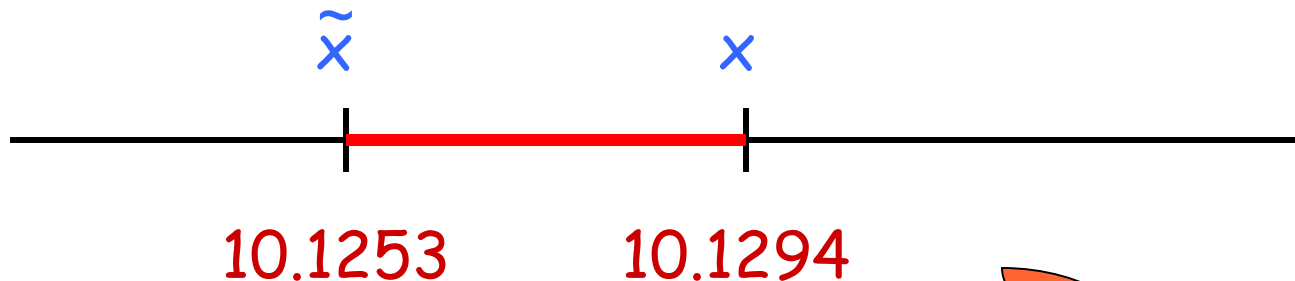
Problema: come si può valutare l'errore commesso da questa approssimazione ?



Idea:

utilizziamo la "*distanza*" tra x e $\tilde{x}$

$$E = |x - \tilde{x}| = |10.1294 - 10.1253| = 0.0041$$



ERRORE ASSOLUTO

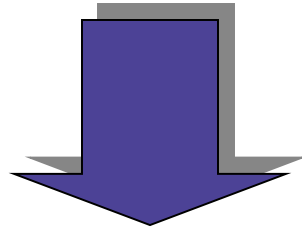


Esempio 1

$$x_1 = 10.\underline{12}94$$

$$\tilde{x}_1 = 10.\underline{12}53$$

$\tilde{x}_1$ e' una approssimazione di x_1
corretta a 2 cifre decimali



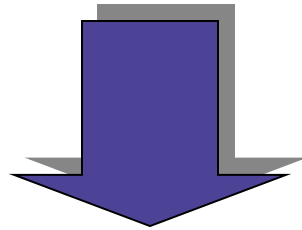
$$E_1 = |x_1 - \tilde{x}_1| = 0.0041 = 0.41 \times 10^{-2} < 10^{-2}$$

Esempio 2

$$x_2 = 0.\underline{0000}326$$

$$\tilde{x}_2 = 0.\underline{0000}215$$

$\tilde{x}_2$ e' una approssimazione di x_2
corretta a 4 cifre decimali



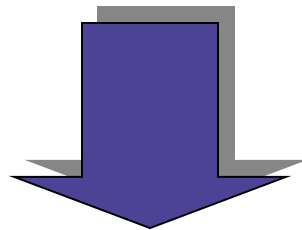
$$E_2 = |x_2 - \tilde{x}_2| = 0.0000111 = 0.111 \times 10^{-4} < 10^{-4}$$

Esempio 3

$$x_3 = 0.\underline{0074132}30$$

$$\tilde{x}_3 = 0.\underline{0074132}96$$

$\tilde{x}_3$ e' una approssimazione di x_3
corretta a 7 cifre decimali

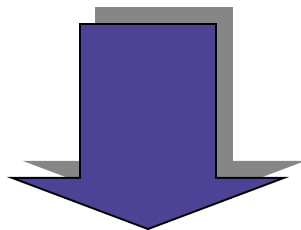


$$E_3 = |x_3 - \tilde{x}_3| = 0.000000066 = 0.66 \times 10^{-7} < 10^{-7}$$

Proprieta' dell'errore assoluto

$\tilde{x}$ e' un'approssimazione di x

corretta ad m cifre decimali

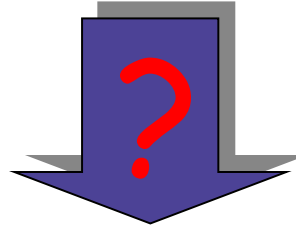


$$E = |x - \tilde{x}| < 10^{-m}$$

MA

Se vale

$$E = |x - \tilde{x}| < 10^{-m}$$



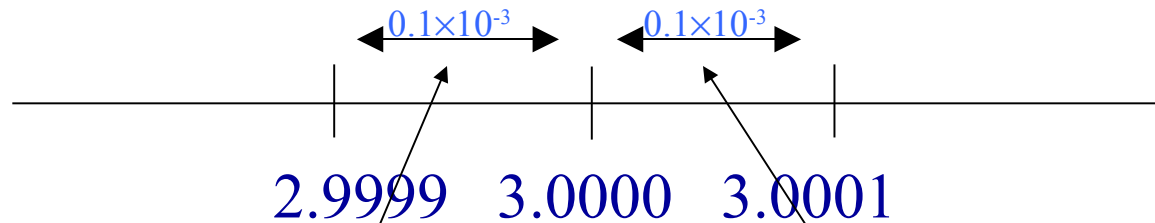
$\tilde{x}$ e' un'approssimazione di x

corretta ad m cifre decimali



È possibile cioè **invertire** la proprietà
dell'errore assoluto?

Esempio



$$x_1 = 2.9999$$

$$x = 3.0000$$

$$x_2 = 3.0001$$

$$E_1 = |x_1 - x| = 0.1 \times 10^{-3}$$

$$E_2 = |x - x_2| = 0.1 \times 10^{-3}$$

In entrambi i casi l'errore assoluto è **lo stesso!**

MA...

$$x = 3.0000$$

$$x_2 = 3.0001$$

♦ x_2 e' un'approssimazione di x corretta a 3 cifre decimali



In questo caso la proprieta'
dell'errore assoluto
si inverte

...MENTRE

$$x_1 = 2.9999$$

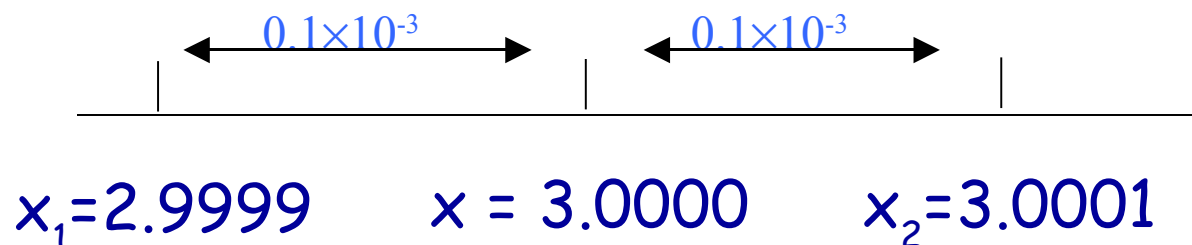
$$x = 3.0000$$

- ♦ x_1 e x non hanno alcuna cifra uguale!



In questo caso la proprietà
dell'errore assoluto
NON si inverte

Cosa hanno di particolare questi numeri ?

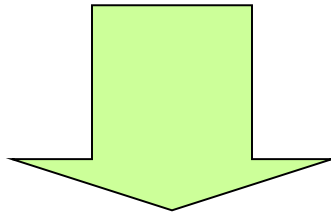


OSSERVAZIONE

Aggiungendo 1 all'ultima cifra di x
si ottiene x_2

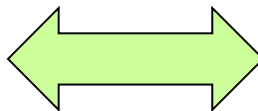
Sottraendo 1 all'ultima cifra di x
si ottiene x_1

L'esempio mostrato è l'unico caso in cui
la proprietà non si inverte,
a causa della rappresentazione posizionale.



Si assume, pertanto:

$$E = |x - \tilde{x}| < 10^{-m}$$



$\tilde{x}$ è un'approssimazione
di x corretta
ad m cifre decimali

ERRORE RELATIVO

Esempio 1

$$x_1 = 10.12$$

$$\tilde{x}_1 = 10.05$$



$$E_1 = |x_1 - \tilde{x}_1| = 0.07$$

$$x_2 = 10000.14$$

$$\tilde{x}_2 = 10000.07$$



$$E_2 = |x_2 - \tilde{x}_2| = 0.07$$

$$E_1 = E_2$$

Ma nel secondo caso l'approssimazione è
intuitivamente migliore

Esempio 2

$$x_1 = 1.1249$$

$$\tilde{x}_1 = 1.1247$$



$$E_1 = |x_1 - \tilde{x}_1| = 0.0002$$

$$x_2 = 130000.5$$

$$\tilde{x}_2 = 130000.3$$



$$E_2 = |x_2 - \tilde{x}_2| = 0.2$$

$$E_1 \ll E_2$$

Ma, ancora una volta, la seconda approssimazione
appare intuitivamente migliore

L'errore assoluto non tiene conto
della grandezza del valore da
approssimare



Riprendiamo l'Esempio 1

$$x_1 = 10.12$$

$$\tilde{x}_1 = 10.05$$

$$x_2 = 10000.14$$

$$\tilde{x}_2 = 10000.07$$



Dividiamo l'errore assoluto per il numero da approssimare...

$$E'_1 = \frac{|x_1 - \tilde{x}_1|}{|x_1|}$$

$$\cong 0.0069 = 0.69 \times 10^{-2}$$

$$E'_2 = \frac{|x_2 - \tilde{x}_2|}{|x_2|}$$

$$\cong 0.0000069 = 0.69 \times 10^{-5}$$

E'_2 è più piccolo di E'_1

Riprendiamo l'Esempio 2

$$x_1 = 1.1249$$

$$\tilde{x}_1 = 1.1247$$

$$x_2 = 130000.5$$

$$\tilde{x}_2 = 130000.3$$



Dividiamo l'errore assoluto per il numero da approssimare...

$$E'_1 = \frac{|x_1 - \tilde{x}_1|}{|x_1|}$$

$$\cong 0.0000017 = 0.17 \times 10^{-5}$$

$$E'_2 = \frac{|x_2 - \tilde{x}_2|}{|x_2|}$$

$$\cong 0.000000015 = 0.15 \times 10^{-7}$$

E'_2 è più piccolo di E'_1

ERRORE RELATIVO



Si misura l'errore utilizzando come
unità di misura
il modulo del valore da approssimare

CIOE'

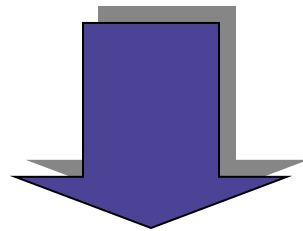
si "scala" il valore assoluto rispetto
al modulo del valore da approssimare

Esempio 1

$$x_1 = \underline{10.12}94$$

$$\tilde{x}_1 = \underline{10.12}53$$

$\tilde{x}_1$ è una approssimazione di x_1 corretta a
4 cifre significative



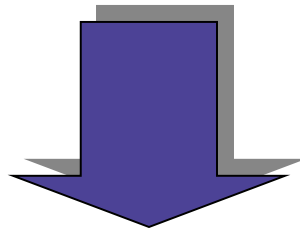
$$E'_1 = 0.0004 = 0.40 \times 10^{-3} < 10^{-3} = 10^{-4+1}$$

Esempio 2

$$x_2 = 0.0000\underline{1}01$$

$$\tilde{x}_2 = 0.0000\underline{1}19$$

$\tilde{x}_2$ è una approssimazione di x_2 corretta a
1 cifra significativa



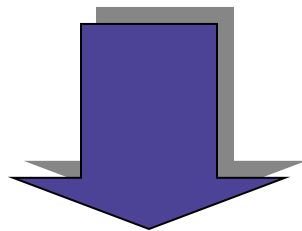
$$E'_2 = 0.18 \times 10^0 < 10^0 = 10^{-1+1}$$

Esempio 3

$$x_3 = \underline{741.3296}$$

$$\tilde{x}_3 = \underline{741.3205}$$

$\tilde{x}_3$ è una approssimazione di x_3 corretta a
5 cifre significative



$$E'_3 \cong 0.000012 = 0.12 \times 10^{-4} < 10^{-4} = 10^{-5+1}$$

Proprieta' dell'errore relativo

$\tilde{x}$ è un'approssimazione di x
corretta a m cifre significative



$$E' = \frac{|x - \tilde{x}|}{|x|} < 10^{-m+1}$$

MA

Se vale

$$E' = \frac{|x - \tilde{x}|}{|x|} < 10^{-m+1}$$

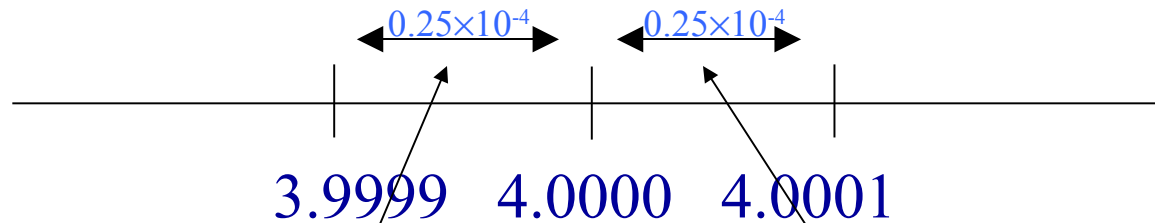


$\tilde{x}$ è un'approssimazione di x
corretta a m cifre significative

?

È possibile cioè **invertire** la proprietà
dell'errore relativo?

Esempio



$$x_1 = 3.9999$$

$$x = 4.0000$$

$$x_2 = 4.0001$$

$$E_1 = \frac{|x_1 - x|}{|x|} = 0.25 \times 10^{-4}$$

$$E_2 = \frac{|x - x_2|}{|x|} = 0.25 \times 10^{-4}$$

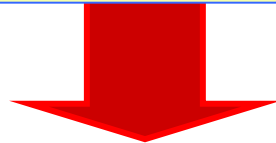
In entrambi i casi l'errore relativo è **lo stesso!**

MA...

$$x = 4.0000$$

$$x_2 = 4.0001$$

♦ x_2 e' un'approssimazione di x corretta a 4 cifre significative (NON 5)



In questo caso la proprieta'
dell'errore relativo
NON si inverte

...ancora

$$x_1 = 3.9999$$

$$x = 4.0000$$

- ◆ x_1 e x non hanno alcuna cifra uguale!



Anche in questo caso la proprietà
dell'errore relativo
NON si inverte

In generale...

La proprietà dell'errore relativo
non si può invertire,
ma si può dire che...

Se

$$E' = \frac{|x - \tilde{x}|}{|x|} < 10^{-m+1}$$

allora

$\tilde{x}$ è un'approssimazione di x
corretta ad almeno
 m
cifre significative

In conclusione

ERRORE
ASSOLUTO

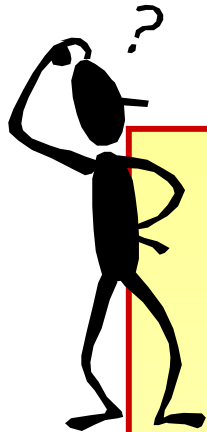


Informazioni sulle
cifre decimali

ERRORE
RELATIVO



Informazioni sulle
cifre significative



Errori

in un sistema aritmetico

a precisione finita

(es. sistema aritmetico di un calcolatore)

ERRORE DI ROUND - OFF

Esempio: $F = (b = 10, t = 3, \dots, \dots)$

$x = 11.246 = 0.11246 \times 10^2$ non è rappresentabile esattamente in F

$fl(x) = 0.112 \times 10^2$

è una approssimazione di x

Rappresentazione floating point normalizzata (f.p.n.) in F

Che errore si commette
approssimando x con $fl(x)$?

$$x = 0.11246 \times 10^2 \quad fl(x) = 0.112 \times 10^2$$



$$E = |x - fl(x)| = 0.46 \times 10^{-3}$$

Errore assoluto di round-off

$$E' = \frac{|x - fl(x)|}{|x|} = 0.41 \times 10^{-2}$$

Errore relativo di round-off



Osservazione

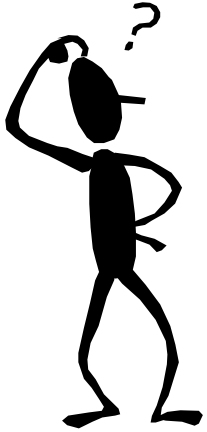
$$E = |x - fl(x)| = 0.46 \times 10^{-3}$$

$fl(x)$ è una approssimazione di x
corretta a al più 3 cifre decimali

$$E' = \frac{|x - fl(x)|}{|x|} = 0.41 \times 10^{-2}$$

$fl(x)$ è una approssimazione di x
corretta ad almeno 2 cifre significative

Nella rappresentazione floating-point normalizzata le
cifre decimali e le cifre significative
coincidono



Qual è il **massimo errore relativo**
che si commette approssimando
 x con $fl(x)$?

Esempio: $b = 10, t = 5$

Sistema con Arrotondamento

$x = f \times b^e$	$fl(x) = \underline{f} \times b^e$	$E' = \frac{ x - fl(x) }{ x }$
0.1111288×10^4 - 0.1152379×10^{-1}	0.111113×10^4 - 0.11524×10^{-1}	0.11×10^{-5} 0.26×10^{-4}
0.5723368×10^2 - 0.4617751×10^1	0.57233×10^2 - 0.46178×10^1	0.18×10^{-6} 0.11×10^{-4}

$$E' \leq 0.5 \times 10^{1-t}$$

Esempio: $b = 10, t = 5$

Sistema con Troncamento

$x = f \times b^e$	$fl(x) = \underline{f} \times b^e$	$E' = \frac{ x - fl(x) }{ x }$
0.1111288×10^4 - 0.1152379×10^{-1}	0.11112×10^4 - 0.11523×10^{-1}	0.79×10^{-5} 0.61×10^{-4}
0.5723378×10^2 - 0.4617751×10^1	0.57233×10^2 - 0.46177×10^1	0.14×10^{-5} 0.11×10^{-4}

$$E' \leq 10^{1-t}$$

In generale, in un sistema a t cifre, per l'arrotondamento

$$x = f \times b^e, \quad (f = 0.m) \qquad \text{fl}(x) = \tilde{f} \times b^e, \quad (\tilde{f} = 0.\tilde{m})$$

◆ $|f| \geq b^{-1}$ (x normalizzato)

◆ $|f - \tilde{f}| \leq 1/2 b^{-t}$



$$\frac{|x - \text{fl}(x)|}{|x|} = \frac{|f - \tilde{f}|}{|f|} \leq \frac{1 b^{-t}}{2 b^{-1}} = \frac{1}{2} \times b^{1-t}$$

$$u = 0.5 \times b^{1-t}$$

massimo errore relativo che si commette
approssimando x con $\text{fl}(x)$ in un
sistema aritmetico floating-point
con base b e precisione t

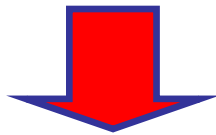
MASSIMA ACCURATEZZA RELATIVA

In generale, in un sistema a t cifre, per il troncamento

$$x = f \times b^e, \quad (f = 0.m) \qquad fl(x) = \tilde{f} \times b^e, \quad (\tilde{f} = 0.\tilde{m})$$

$$\diamond \quad | \underset{\sim}{f} | \geq b^{-1} \quad (x \text{ normalizzato})$$

$$\diamond \quad | f - \tilde{f} | \leq b^{-t}$$



$$\frac{|x - fl(x)|}{|x|} = \frac{|f - \tilde{f}|}{|f|} \leq \frac{b^{-t}}{b^{-1}} = b^{1-t}$$

$$u' = b^{1-t}$$

MASSIMA ACCURATEZZA RELATIVA

Esempi

1) $b = 10$ $t = 5$

$$u = 0.5 \times 10^{-4} \quad (\text{arrotondamento})$$
$$u' = 10^{-4} \quad (\text{troncamento})$$

2) Sistema aritmetico standard IEEE
 $b = 2$, arrotondamento

Singola precisione: $t = 23$

$$u = 0.5 \times 2^{-23+1} = 2^{-23} \cong 0.119 \times 10^{-6}$$

Doppia precisione: $t = 52$

$$u = 0.5 \times 2^{-52+1} = 2^{-52} \cong 0.222 \times 10^{-15}$$

OSSERVAZIONE

Posto

si ha $\delta = \frac{fl(x) - x}{x}$

$$fl(x) = x + x\delta = x(1 + \delta)$$

Con

$$E' = |\delta| \leq u = 0.5 b^{1-t} \quad \text{arrotondamento}$$

$$E' = |\delta| \leq u' = b^{1-t} \quad \text{troncamento}$$

ERRORE DI ROUND-OFF NELLE OPERAZIONI ARITMETICHE

Esempio

$$F = (b = 10, t = 2, E_{\min} = -5, E_{\max} = 5)$$

$$x = 18 \in F$$

$$fl(x) = 0.18 \times 10^2 \in F$$

$$y = 95 \in F$$

$$fl(y) = 0.95 \times 10^2 \in F$$

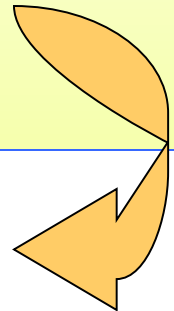
$$x + y = 113 = 0.113 \times 10^3$$

$$x + y \notin F$$

$$r = x \# y \quad x, y \in F \Rightarrow x = fl(x), y = fl(y)$$

In generale il risultato di un'operazione
aritmetica $\#$ tra numeri macchina
non è un numero macchina

ERRORE DI ROUND-OFF



Esempio

$$F = (b = 10, t = 4, E_{\min} = -5, E_{\max} = 5)$$

$$x = fl(x) = 0.1035 \times 10^3 \in F, y = fl(y) = 0.7281 \times 10^1 \in F$$

$$fl(x) \ominus fl(y) = ?$$

Operazione eseguita in F



Come calcolare $fl(x) \ominus fl(y)$?

$$x = fl(x) = 0.1035 \times 10^3 \quad y = fl(y) = 0.7281 \times 10^1$$

+	1	0	3	5		+	3
---	---	---	---	---	--	---	---

+	7	2	8	1		+	1
---	---	---	---	---	--	---	---

In aritmetica "classica" si dispongono
i numeri *in colonna* e si sottraggono

$$x + y = 0.1035 \times 10^3 - 0.7281 \times 10^1 =$$

$$\begin{array}{r} 103.500 \\ - 7.281 \\ \hline 96.219 \end{array} = 0.96219 \times 10^2$$

In aritmetica floating point



Si sommano le mantisse
dopo aver ricondotto gli addendi
allo stesso esponente

1° PASSO

Calcolo della differenza degli ordini di grandezza dei due addendi

$$\begin{array}{l} x = 0.1035 \times 10^3 \\ y = 0.7281 \times 10^1 \end{array} \quad \longrightarrow \quad + 3 - 1 = 2$$

2° PASSO

"Shift" di **2 cifre** della mantissa di **y** per ottenere una rappresentazione con esponente uguale a quello di **x**

$$\begin{array}{l} x = 0.1035 \times 10^3 \\ y = 0.7281 \times 10^1 \end{array} \quad \longrightarrow \quad \begin{array}{l} x = 0.1035 \times 10^3 \\ y_{\text{shift}} = 0.0072 \times 10^3 \end{array}$$

3° PASSO Differenza delle mantisse

$$\begin{array}{lcl} x = & 0.1035 & \times 10^3 \\ y_{\text{shift}} = & 0.0072 & \times 10^3 \end{array} \longrightarrow x - y_{\text{shift}} = 0.0963 \times 10^3$$

4° PASSO Normalizzazione del risultato

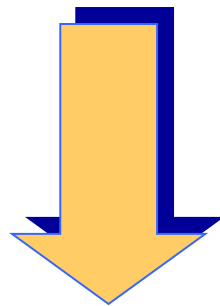
$$x \ominus y = \text{fl}(x) \ominus \text{fl}(y) = 0.9630 \times 10^2$$

Osservazione

$$x - y = 0.96219 \times 10^2$$

mentre

$$fl(x) \ominus fl(y) = 0.9630 \times 10^2 \quad fl(x - y) = 0.9622 \times 10^2$$



$$fl(x) \ominus fl(y) \neq fl(x - y)$$

E' possibile fare in modo che

$$\text{fl}(x) \ominus \text{fl}(y) = \text{fl}(x - y)$$

?

$$fl(x) \ominus fl(y) \neq fl(x - y)$$

A causa della perdita di
alcune cifre
significative di y nel 2° passo
(shift della mantissa)



Strategia

Utilizzare per gli operandi e per i risultati intermedi locazioni di memoria (registri dell'unità aritmetica) di lunghezza $t > t$

Esempio

$$t = 4 \quad x = 0.1035 \times 10^3 \quad y = 0.7281 \times 10^1$$

$fl(x) \ominus fl(y)$ eseguita con registri di lunghezza

$$t_{reg} = 2t = 8$$

1° PASSO

$$3 - 1 = 2$$

2° PASSO

y

+	7	2	8	1	+	1
---	---	---	---	---	---	---

Memoria centrale

$Y_{shift} \equiv y$

+	0	0	7	2	8	1	0	0	+	3
---	---	---	---	---	---	---	---	---	---	---

Registro

3° PASSO

x

+	1	0	3	5	0	0	0	0	+	3
---	---	---	---	---	---	---	---	---	---	---

y

+	0	0	7	2	8	1	0	0	+	3
---	---	---	---	---	---	---	---	---	---	---

x - y

+	0	9	6	2	1	9	0	0	+	3
---	---	---	---	---	---	---	---	---	---	---

4° PASSO

Arrotondamento e normalizzazione

x - y

+	0	9	6	2	1	9	0	0	+	3
---	---	---	---	---	---	---	---	---	---	---



+	9	6	2	2	+	2
---	---	---	---	---	---	---

Registro

Memoria centrale

$$fl(x \cdot y) = fl(x) \odot fl(y)$$

F sistema aritmetico f.p. ha
massima accuratezza dinamica se

$$x, y \in F \Rightarrow fl(x) \oplus fl(y) = fl(x \# y)$$

L'errore relativo di round-off introdotto da una
operazione aritmetica f.p. a precisione finita
e' uguale all'errore di
round-off di **rappresentazione** del risultato della
corrispondente operazione in aritmetica a precisione infinita

CIOE'

$$fl(x) \oplus fl(y) = fl(x \# y) = (x \# y)(1 + \delta), \quad |\delta| \leq u$$

Epsilon macchina

Esempio

$$b = 10$$

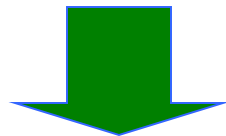
$$t = 3$$

$$t_{\text{reg}} = 6$$

$$1 = 0.100 \times 10^1$$

$$x = 0.325 \times 10^{-2}$$

$$\begin{aligned} 1 \oplus x &= 0.100000 \times 10^1 \oplus 0.\underline{000}325 \times 10^1 = \\ &= 0.100 \times 10^1 \end{aligned}$$



$$1 \oplus x = 1$$

In un sistema aritmetico f.p. a precisione finita **esistono numeri macchina non nulli** che si comportano come lo zero nella somma f.p. con 1

Si definisce **epsilon macchina** il più piccolo numero macchina ε tale che:

$$1 \oplus \varepsilon = \text{fl}(1 + \varepsilon) > 1$$

Esempio

$$b = 10$$

$$t = 3$$

$$t_{\text{reg}} = 6$$

Si vuole calcolare η massimo numero
macchina tale che $1 \oplus \eta = 1$

Posto $\eta = 0.d_1d_2d_3 \times 10^p$ con $1 = p + 3$, si ha

1		0.	1	0	0	0	0	0	$\times 10^1$	+
η		0.	0	0	0	d_1	d_2	d_3	$\times 10^{p+3}$	=
$1 \oplus \eta$		0.	1	0	0	d_1	d_2	d_3	$\times 10^{p+3}$	

$$1 \oplus \eta = 1 \iff d_1 < 5,$$

$\eta = 0.499 \times 10^{-2}$ max numero macchina
tale che $1 \oplus \eta = 1$



$\varepsilon = 0.500 \times 10^{-2} = \frac{1}{2} b^{1-t}$ **ε -macchina**

In un sistema aritmetico f.p.
con base β e precisione finita t ,
con massima accuratezza dinamica
si ha:

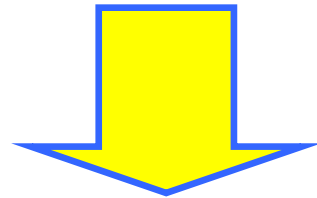
$$\varepsilon = \frac{1}{2} b^{1-t} = u$$

ε -macchina

max accuratezza
relativa

$$\varepsilon = 1/2 b^{1-t}$$

Nota la base b ,
conoscere ε equivale a
conoscere t



L' ε -macchina fornisce informazioni sulla
precisione del sistema aritmetico f.p.

Implementazione dell'algoritmo su una
workstation IBM RISC System/6000
(sistema aritmetico standard IEEE, $\beta = 2$)

Singola precisione: $t = 23$ (in base 2)

$$u = 2^{-23} = \frac{1}{2} 2^{-23+1} \cong 0.119 \times 10^{-6} \Rightarrow$$

$$\Rightarrow t \approx 7 \text{ (in base 10)}$$

Doppia precisione: $t = 52$ (in base 2)

$$u = 2^{-52} = \frac{1}{2} 2^{-52+1} \cong 0.222 \times 10^{-15} \Rightarrow$$

$$\Rightarrow t \approx 16 \text{ (in base 10)}$$

Esempio

$$\beta = 10$$

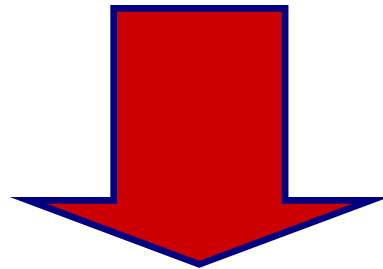
$$t = 4$$

$$t_{\text{reg}} = 2t = 8$$

$$x = 0.1431 \times 10^5$$

$$y = 0.2356 \times 10^1$$

$$\begin{aligned} x \oplus y &= 0.14310000 \times 10^5 \oplus 0.00002356 \times 10^5 = \\ &= 0.1431 \times 10^5 = x \end{aligned}$$



$$x \oplus y = x$$

In generale,

$$y = 0.y_1 y_2 y_3 y_4 \times 10^q$$

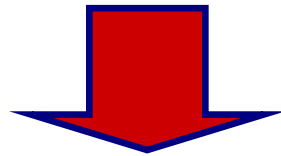
e

$$x = 0.x_1 x_2 x_3 x_4 \times 10^p$$

e $q=p-4$ si ha

$$\begin{array}{r|cccccccccc} x & 0. & x_1 & x_2 & x_3 & x_4 & 0 & 0 & 0 & 0 & \times 10^p & + \\ y & 0. & 0 & 0 & 0 & 0 & y_1 & y_2 & y_3 & y_4 & \times 10^{q+4} & = \\ \hline x \oplus y & 0. & x_1 & x_2 & x_3 & x_4 & y_1 & y_2 & y_3 & y_4 & \times 10^{q+4} \end{array}$$

$$x \oplus y = x \quad \Leftrightarrow \quad y_1 < 5$$



$$y < 0.5 \times 10^{p-4} = 0.5 \times 10^{1-4} \times 10^{p-1} = 1/2 \beta^{1-t} \times 10^{p-1} \leq \varepsilon |x|$$

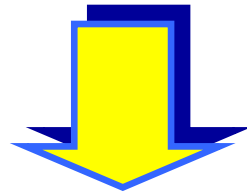
minimo numero macchina tale che $x \oplus y > x$

Nell'esempio precedente:

$$x = 0.1431 \times 10^5$$

$$y = 0.2356 \times 10^1$$

$$y = 0.2356 \times 10^1 < 0.5 \times 10^{-3} \cdot 0.1431 \times 10^5 = \\ = 0.47155 \times 10^2$$



$$x \oplus y = x$$

Problema

calcolo di una somma con un grande numero di addendi:



$$S_1 = \frac{1}{2}$$

$$S_2 = \frac{1}{2} + \frac{1}{4}$$

$$S_3 = \frac{1}{2} + \frac{1}{4} + \frac{1}{8}$$

.....

$$S_n = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots + \frac{1}{2^n}$$

n MOLTO GRANDE!

Problema

calcolo di una somma con un grande numero di addendi:



$$S_1 = \frac{1}{2}$$

$$S_2 = \underbrace{\frac{1}{2}}_{S_1} + \frac{1}{4}$$



$$S_2 = S_1 + \frac{1}{2^2}$$

$$S_3 = \underbrace{\frac{1}{2} + \frac{1}{4}}_{S_2} + \frac{1}{8}$$



$$S_3 = S_2 + \frac{1}{2^3}$$

.....

.....

$$S_n = \underbrace{\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots}_{S_{n-1}} + \frac{1}{2^n}$$



$$S_n = S_{n-1} + \frac{1}{2^n}$$

n MOLTO GRANDE!

n MOLTO GRANDE...

$n=1000$

?

$n=10.000$

?

$n=1.000.000$

?



eseguendo i calcoli in $F(b = 10, t = 7)$ si ha:

n	S_n	$1/2^n$
1.	0.5000000	0.5000000000000000
2.	0.7500000	0.2500000000000000
3.	0.8750000	0.1250000000000000
4.	0.9375000	0.0625000000000000
5.	0.9687500	0.0312500000000000
		
22.	0.9999997	0.00000023841858
23.	0.9999998	0.00000011920929
24.	0.9999999	0.00000005960464
25.	0.9999999	0.00000002980232
26.	0.9999999	0.00000001490116
27.	0.9999999	0.00000000745058

} $1/2^n$ coincide con lo zero di F !

DOPO SOLO $n=25$ S_{n-1} non cambia più!
 $(1/2^n$ non dà più contributo alla somma con $S_{n-1})$

In un sistema aritmetico a precisione finita

Soluzione "naturale":

Quando $\frac{1}{2^n}$ non da' contributo nella somma con S_{n-1} :

$$S_n = S_{n-1} \oplus \frac{1}{2^n} = S_{n-1}$$



Determinare
il piu' piccolo n tale che:

$$\frac{1}{2^n} < \varepsilon \cdot S_{n-1}$$

IN GENERALE

In un processo
numerico
del tipo

$$S_n = S_{n-1} + y_n$$

Si utilizza il *criterio di arresto naturale*

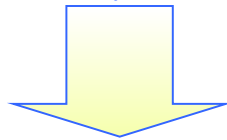
$$|y_n| < \varepsilon \cdot |S_{n-1}|$$

Osservazione

$$b = 10 \quad t = 4 \quad E_{\min} = -9 \quad E_{\max} = 9$$

$$x = 0.2356 \times 10^{-8}$$

$$\begin{aligned} \varepsilon_x &= \varepsilon \cdot |x| = 0.5 \times 10^{-3} \cdot 0.2356 \times 10^{-8} = \\ &= 0.1178 \times 10^{-10} < 0.1 \times 10^{-9} = \text{rmin} \\ &\text{(rmin = minimo numero macchina reale positivo)} \end{aligned}$$



$\varepsilon \cdot |x|$ **provoca underflow**

In questo caso
si pone: $\varepsilon_x = \text{rmin}$

Fine Lezione

