

NUMERI REALI COSTITUISCONO UN INSIEME FINITO E CONTINUO CHE INCLUDE TUTTI I NUMERI razionali e irrazionali. Tra due numeri reali esiste sempre infiniti altri numeri, creando una densità perfetta senza "buchi" o discontinuità. La natura infinita dei numeri reali è matematicamente perfetti ma computazionalmente problematici.

NUMERI FINITI RAPPRESENTANO L'ADATTAMENTO NECESSARIO DEI NUMERI REALI ALLE LIMITAZIONI FISICHE DEI CALCOLATORI (MACCHINE DISCRETE). LA CONVERSIONE DI UN NUMERO REALE ALLA SUA RAPPRESENTAZIONE IN VIRGOLA MOBILE (FLOATING POINT) CON PRECISIONE FINITA È:

$$n \in \mathbb{R} \rightarrow d \in \mathbb{R}$$

[Es. 1] Qual'è la base di d ? (Nota: non sempre è nota) $d = n^o$ "normalizzato" in una forma standard

$$\hookrightarrow d = 21,37 \rightarrow (+ 21.37)_{10} - \text{Forma Scientifica} \rightarrow d = (+ 0.2137)_{10} \cdot 10^2 \xrightarrow{\text{ESponente della base}} [P]$$

[Es. 2]

$$\hookrightarrow d = 0.0045 \rightarrow (+ 0.0045)_{10} - \text{Forma Scientifica} \rightarrow d = (+ 0.45)_{10} \cdot 10^{-3} \xrightarrow{\text{[P]}}$$

Si illustra come qualsiasi numero reale (che potrebbe avere infinite cifre) viene FORZATO in una rappresentazione con:

- MANTISSA FINITA (nº limitato di cifre significative)
- ESPOENTE FINITO (range limitato)

$$d = \pm (0. \underbrace{d_1 d_2 d_3 \dots}_{\text{CIFRE NUMERO}})_\beta \cdot \beta^p$$

$$0 \leq d_1, d_2, d_3, \dots \leq \beta-1 \quad d_1 \neq 0$$

$$\left. \begin{aligned} d &= \pm \underbrace{(d_1 \cdot \beta^{-1} + d_2 \cdot \beta^{-2} + d_3 \cdot \beta^{-3} + \dots)}_{\text{MANTISSA}} \beta \cdot \beta^p \\ &\equiv (M)_\beta \cdot \beta^p, \quad 0 \leq d_i \leq \beta-1, \quad d_1 \neq 0 \end{aligned} \right\} M = \sum_{i=1}^{\infty} d_i \cdot \beta^{-i}$$

NUMERI FINITI SONO IL COMPROMESSO TRA L'INFINITO MATEMATICO E LA FINITETÀ COMPUTAZIONALE.

NUMERI FINITI SI INDICANO CON LA LETTERA \mathbb{F} E SONO UN SOTTOinsieme di \mathbb{R} ($\mathbb{F} \subset \mathbb{R}$).

LA CARATTERIZZAZIONE DI \mathbb{F} O PARAMETRIZZAZIONE DI \mathbb{F} È LA SEGUENTE:

$$\text{DEF: } \mathbb{F}(B, t, \lambda, w)$$

BASE
ESPOENTE PIÙ PICCOLO (MINIMO)
ESPOENTE PIÙ GRANDE (MAXIMO)
nº DI CIFRE CON LE QUALI SCRIVIAMO LA MANTISSA

È l'insieme di n° FINITO:

$$\left\{ \emptyset \right\} \cup \left\{ d \in \mathbb{R}; d = \pm m_t \cdot \beta^p \text{ con } d_1 \neq 0, \lambda \leq p \leq w \right\}$$

$$m_t = \sum_{i=1}^t d_i \cdot \beta^{-i} \quad \beta \geq 2, t > 1$$

t INDICA LA PRECISIONE, IL n° DI CIFRE

Questa è la FORMA STANDARD PER RAPPRESENTARE LA MANTISSA IN UN SISTEMA FLOATING POINT NORMALIZZATO (ovvero, $0.d_1 d_2 d_3 \dots$ è NON IN ALTRI MODI; es. $0.125 \cdot 2^3 = 0.25 \cdot 2^2 = 0.5 \cdot 2^1 = 1.0 \cdot 2^0$).

ESEMPIO: dato l'insieme dei numeri finiti $\mathbb{F}(2, 3, -1, 2)$, vogliamo capire da quanti numeri è formato l'insieme. I numeri saranno a tre cifre:

con la DEFINIZIONE formale di un numero floating point normalizzato in base 2, quindi con la DEFINIZIONE: $\tilde{x} = \pm (0.d_1 d_2 d_3)_2 \cdot 2^p, -1 \leq p \leq 2$ con $d_1 \neq 0 \rightarrow \tilde{x} = \pm (0.1 d_2 d_3)_2 \cdot 2^p$ possiamo scrivere tutti i numeri di questo insieme nel seguente modo:

$\Rightarrow 2 \text{ POSSIBILITA'}$
 $\Rightarrow 4 \text{ POSSIBILITA'}$
 $\tilde{d} = \pm (0, d_1 d_2 d_3)_2 \cdot 2^P$
 (NORMALIZZAZIONE) $\Rightarrow 2 \text{ POSSIBILITA' CIASCUNO } (n \cdot n)$
 $i d_1, d_2 \in \{0, 1\}$
 $n = 2 \cdot 2$

$\overbrace{32}^{2 \cdot 1 \cdot 2 \cdot 2 \cdot 4 + 1} = 33 \text{ n° DISTINTI}$

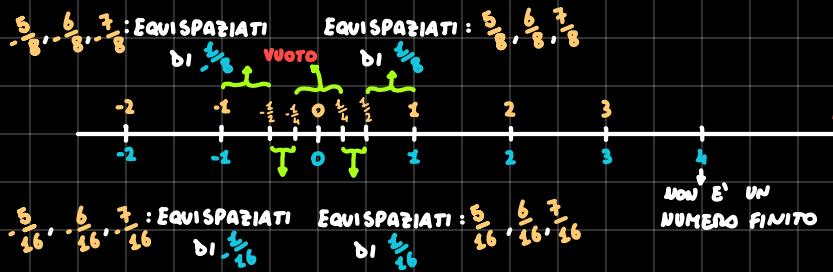
INDICA LO ZERO CHE NON PUO' ESSERE RAPPRESENTATO
 IN FORMA NORMALIZZATA ($d_1 \neq 0$)

PER COME SONO DISTRIBUITI SULL' ASSE REALE SEGRETE IL PROCEDIMENTO:

$0.100 = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 0 \cdot 2^{-3} = \frac{1}{2} = \frac{4}{8}$	$\frac{4}{8} \cdot 2^{-1} = \frac{4}{16}$	$\frac{4}{8} \cdot 2^0 = \frac{4}{8}$	$\frac{4}{8} \cdot 2^1 = \frac{4}{4}$	$\frac{4}{8} \cdot 2^2 = \frac{4}{2}$
$0.101 = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = \frac{1}{2} + \frac{1}{8} = \frac{5}{8}$	$\frac{5}{8} \cdot 2^{-1} = \frac{5}{16}$	$\frac{5}{8} \cdot 2^0 = \frac{5}{8}$	$\frac{5}{8} \cdot 2^1 = \frac{5}{4}$	$\frac{5}{8} \cdot 2^2 = \frac{5}{2}$
$0.110 = 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} = \frac{1}{2} + \frac{1}{4} = \frac{6}{8}$	$\frac{6}{8} \cdot 2^{-1} = \frac{6}{16}$	$\frac{6}{8} \cdot 2^0 = \frac{6}{8}$	$\frac{6}{8} \cdot 2^1 = \frac{6}{4}$	$\frac{6}{8} \cdot 2^2 = \frac{6}{2}$
$0.111 = 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} = \frac{7}{8}$	$\frac{7}{8} \cdot 2^{-1} = \frac{7}{16}$	$\frac{7}{8} \cdot 2^0 = \frac{7}{8}$	$\frac{7}{8} \cdot 2^1 = \frac{7}{4}$	$\frac{7}{8} \cdot 2^2 = \frac{7}{2}$

$\left. \begin{array}{l} \\ \\ \\ \end{array} \right\} + (\text{E LO ZERO})$

QUESTA RAPPRESENTAZIONE FLOATING POINT MOSTRA LA TRASFORMAZIONE SISTEMATICA DA NOTAZIONE BINARIO POSIZIONALE (IL VALORE DI OGNI CIFRA DIPENDE DALLA SUA POSIZIONE $[2^{-1}, 2^{-2}, 2^{-3}]$) NELLA SEQUENZA NUMERICA) A VALORE DECIMALE. I NUMERI SONO DISTRIBUITI SULL'ASSE:



LA RAPPRESENTAZIONE GRAFICA SULL'ASSE MOSTRA COME I VUOTI TRA I NUMERI FLOATING POINT CRESCANO ESPONENZIALMENTE ALLONTANANDO SI DA ZERO CREANDO UNA "DENSITA' NON UNIFORME" CHE CONTRADICE L'INTUZIONE MATEMATICA DEL CONTINUO. QUESTO EVIDENZIA IL CONCETTO DI "APPROXIMAZIONE COMPUTAZIONALE": OGNI CALCOLO NUMERICO NON OPERA SUL VERO CONTINUO REALE, MA SU QUESTA GRIGLIA FINITA E IRREGOLARE DI VALORI RAPPRESENTABILI. NELLA GRIGLIA SI NOTI COME I NUMERI SIANO ADDENSATI VICINO ALLO ZERO E LONTANI TRA LORO DISTANTI DA ZERO.

STANDARD ANSI/IEEE 754 (INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS - 754 NUMERO 10. DELLO STANDARD)

E' UN PROTOCOLLO INTERNAZIONALE CHE DEFINISCE IN MODO UNIFORME COME I NUMERI REALI VENGANO RAPPRESENTATI E MANIPOLATI IN TUTTI I SISTEMI INFORMATICI MODERNI. QUESTO STANDARD GARANTISCE CHE UN CALCOLO FLOATING POINT PRODUCA GLI STESSI RISULTATI IDENTICI SU QUALESiasi COMPUTER, PROCESSORE O LINGUAGGIO DI PROGRAMMAZIONE, ELIMINANDO LE INCOMPATIBILITA'.

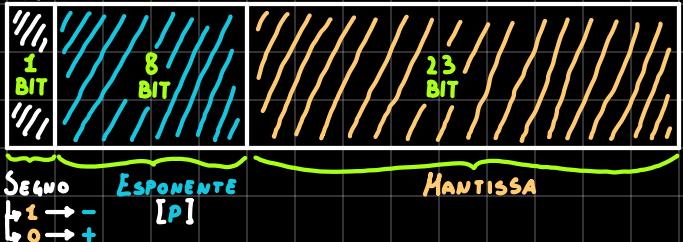
OLTRE A STANDARDIZZARE I FORMATI DI RAPPRESENTAZIONE (32 BIT, 64 BIT) DEFINISCE ANCHE LE REGOLE PER L'ARROTONDAMENTO, LA GESTIONE DEGLI ERRORI E VALORI SPECIALI COME INFINITO E NaN (NOT A NUMBER), ASSICURANDO LA PORTABILITA' DEL SOFTWARE (CODICE MACCHINA) E RIPRODUCIBILITA' DEI RISULTATI SU SCALA GLOBALE.

LO STANDARD DEFINISCE QUATTRO FORMATI FLOATING POINT IN DUE GRUPPI, EXTENDED (CHE NON TRATTEREMO) ED BASIC, CIASCUÑO CON DUE PRECISIONI, SINGLE E DOUBLE.

VEDIAMO IL BASIC SINGLE E BASIC DOUBLE.

BASIC SINGLE (32 Bit) : $\text{IF}(2, 24, -126, +127)$

AREA DI MEMORIA



CAMPARI:

- 1 BIT PER IL SEGNO
- 8 BIT PER L'ESPOLENTE CON BIAS. IL BIAS TRASFORMA TUTTI GLI ESPOLENTE IN NUMERI POSITIVI, SEMPLIFICANDO L'HARDWARE E PERMETTENDO CONFRONTI DIRETTI TRA NUMERI F. POINT.
- DEF: È UN NUMERO (COSTANTE FISSA) CHE SI AGGIUNGE, PRIMA DI MEMORIZZARE L'ESPOLENTE, PER SPOSTARE TUTTI I VALORI IN UN RANGE (POSITIVO) CONVENIENTE.

Bias : 127 (COSTANTE FISSA) (zero) (NaN o ∞)

Esponente Reale : da -126 a +127 (ESTREMI RISERVATI)

Esponente Memorizzato : da 1 a 254 (SEMPRE POS.)

L'Esponente p : $-126 \leq p \leq +127$ e $\tilde{p} = p + 127 \in [0, 2047]$ (NaN o ∞)

• 23 BIT PER LA MANTISSA, IL BIT 1 È IMPLICITO (SOTTOINTESA)

NUMERI VICINO ALLO ZERO

$00\ldots0 : 0 \rightarrow -127$ } 8 BIT (VALORI DA 0 - 255) MA 0 E 255 SONO RISERVATI PER VALORI SPECIALI (ZERO, INFINITO, NaN), RANGE UTILIZZABILE : 1 A 254

NUMERI LONTANI DALLO ZERO

VALE L'ANALOGO RAIONAMENTO PER IL DOUBLE

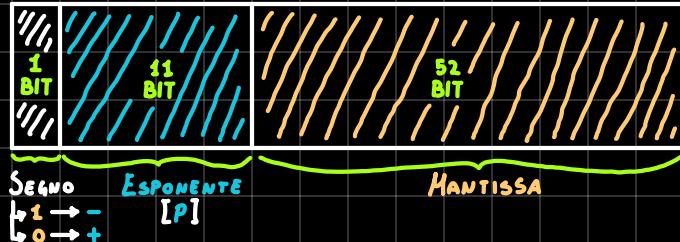
IL VALORE v DI UN n $\text{fl}(d)$ È :

- SE $\tilde{p} = 255$ ED $m \neq 0$ ALLORA $v = \text{NaN}$ (Not a Number)
- SE $\tilde{p} = 255$ ED $m = 0$ ALLORA $v = (-1)^s \infty$ (INFINITY) ($s = 0 \rightarrow +\infty$ $s = 1 \rightarrow -\infty$)
- SE $0 < \tilde{p} < 255$ ALLORA $v = (-1)^s \cdot (1.m) \cdot 2^{\tilde{p}-127}$ (NORMALIZED) → (ASO PIÙ RICORRENTE (QUELLO PIÙ RIPETUTO))
- SE $\tilde{p} = 0$ ED $m \neq 0$ ALLORA $v = (-1)^s \cdot (1.m) \cdot 2^{-127}$ (NOT NORMALIZED)
- SE $\tilde{p} = 0$ ED $m = 0$ ALLORA $v = (-1)^s \cdot 0$ (Zero)

DATO $\text{IF}(B, t, \lambda, w)$ LO STANDARD CONSIDERA LA MANTISSA NELLA FORMA : $m_t = d_0.d_1d_2\dots d_t$, $d_0 \neq 0$ E MEMORIZZA SOLO $d_1\dots d_t$, COMPORTANDO UNA DIFFERENZA SULL'ESPOLENTE RISPETTO ALLE NOTAZIONI INTRODOTTE PRIMA. L'ESPOLENTE È MEMORIZZATO IN FORMA BIASED (PER TRASLAZIONE). (10^6), NEGLI NOTAZIONI CLASSICHE, SE SI PENSA ALLA rappresentazione SCIENTIFICA IN BASE 10, NORMALMENTE SCRIVERESTI UN NUMERO IN FORMA : $m_t \cdot B^p (0.d_1\dots d_t)_2 \cdot 2^p$. IN QUELLE DELLO STANDARD IEEE 754, SFRUTTA IL FATTO CHE NEI NUMERI NORMALIZZATI, $d_0 = 1$ SEMPRE. PER NON "SPRECARE" UN BIT NON LO MEMORIZZA; INVECE DI SALVARE $1.d_1d_2\dots d_t$ SALVA SOLO 1 BIT DOPO IL PUNTO ($d_1d_2\dots d_t$). (ASI GUADAGNA UN BIT DI PRECISIONE IN PIÙ).

BASIC DOUBLE (64 Bit) : $\text{IF}(2, 53, -1022, +1023)$

AREA DI MEMORIA



CAMPARI:

- 1 BIT PER IL SEGNO
- 11 BIT PER L'ESPOLENTE CON BIAS.

Bias : 1023 (COSTANTE FISSA) (zero) (NaN o ∞)

Esponente Reale : da -1022 a +1023 (ESTREMI RISERVATI)

Esponente Memorizzato : da 1 a 2046 (SEMPRE POS.)

L'Esponente p : $-1022 \leq p \leq +1023$ e $\tilde{p} = p + 1023 \in [0, 2047]$ (NaN o ∞)

• 52 BIT PER LA MANTISSA, IL BIT 1 È IMPLICITO (SOTTOINTESA)

Un numero NORMALIZZATO È QUELLO IN CUI L'ESPOLENTE MEMORIZZATO NON È 0 E NON È TUTTI 1. Nei numeri NORMALIZZATI SI SFRUTTA IL FATTO CHE LA MANTISSA È SEMPRE DEL TIPO : 1. MANTISSA IN BASE 2, CIOÈ CI È SEMPRE UN BIT IMPLICITO 1 DAVANTI. CHE AUMENTA LA PRECISIONE SENZA DOVERLO MEMORIZZARE

- NELLA NOTAZIONE CLASSICA CON MANTISSA $m_t = d_0.d_1 \dots d_t$ IL VAORE E' $v = m_t \cdot \beta^p$
- NELLA NOTAZIONE DELLO STANDARD, VISTO CHE NON SALVA d_0 , DEVE RICOSTRUIRE IL n^o COME: $v = (1.d_1 \dots d_t)_\beta \cdot \beta^p$
- LA SOTTRAZIONE DEL BIT IMPLICATO FA SI' CHE L'ESPOENTE MEMORIZZATO VENNA GESTITO DIVERSAMENTE RISPETTO ALLE RAPPRESENTAZIONI PRECEDENTI.
- LO STANDARD INTRODUCE IL **BIAS**, E L'ESPOENTE MEMORIZZATO NON E' p MA $\tilde{p} = p + \text{BIAS}$.

- NORMALIZZATI:** $v = (-1)^s \cdot (1.d_1 \dots d_t) \cdot 2^p$
- BIT IMPLICATO 1 SEMPRE
- ESPOENTE MEMORIZZATO: $\tilde{p} = p + \text{BIAS}$, $\tilde{p} \neq 0$ $V \tilde{p} \neq 255$
- MASSIMA PRECISIONE POSSIBILE PERCHE' SFUORI TUTTI I BIT DELLA MANTISSA
- INTERVALLO DI VALORI: DAL PIU' PICCOLI FINO AI PIU' GRANDI, MA NON SI ARRIVA MAI A ZERO.
- MINIMO NORMALIZZATO: $1.0 \cdot 2^{-226} \approx 1.18 \cdot 10^{-38}$
- DENORMALIZZATI:** $v = (-1)^s \cdot (0.d_1 \dots d_t) \cdot 2^{p-\text{BIAS}}$
- ESPOENTE MEMORIZZATO UGUALE A 0
- NON C'E' IL BIT IMPLICATO
- MANTISSA: $0.d_1 \dots d_t$
- ESPOENTE REALE: $p = \tilde{p} - \text{BIAS}$
- NON ESISTE IL BIT IMPLICATO 1, I NUMERI POSSONO ESSERE MOLTO PIU' PICCOLI DEL MINIMO NORMALIZZATO PERMETTENDO IL GRADUALE UNDERFLOW.
- MINIMO DENORMALIZZATO: $0.000 \dots 01 \cdot 2^{-226} \approx 1.45 \cdot 10^{-45}$

FORMULA PER CALCOLARE L'ESPOENTE CON BIAS NEI SISTEMI FLOATING POINT:

$$\begin{aligned} e &= p - \lambda & \left\{ \begin{array}{l} e: \text{ESPOENTE MEMORIZZATO (EFFETTIVAMENTE SALVATO NEI BIT)} \\ p: \text{ESPOENTE EFFETTIVO (MATEMATICO) CHE VOGLIAMO RAPPRESENTARE} \end{array} \right. \\ p &= e + \lambda & \lambda: \text{BIAS (VALORE DI POLARIZZAZIONE)} \end{aligned}$$

$e = p - \lambda \rightarrow$ ESPOENTE MEMORIZZATO = ESPOENTE EFFETTIVO - BIAS

$p = e + \lambda \rightarrow$ ESPOENTE EFFETTIVO = ESPOENTE MEMORIZZATO + BIAS

ES: VOGLIAMO RAPPRESENTARE 2^3 ($p=3$, $\lambda=127$), QUINDI $e = 3 + 127 = 130$ (QUESTO VIENE MEMORIZZATO)

DATO a COME SI PASSA AL SUO RAPPRESENTANTE \tilde{a} ? ESEMPIO IN BASE 10 (DECIMALE) $\tilde{a} = (10,5,-50,49)$

DATO $a = 0.345678 \cdot 10^0 \in \mathbb{R}$ PASSARE AL SUO RAPPRESENTANTE \tilde{a} :

Domanda 1) E' IN NOTAZIONE SCIENTIFICA? (SI)

Domanda 2) L'ESPOENTE E' NEL RANGE? (SI)

Domanda 3) MANTISSA $t < 5$? (NO)

SICOME LA MANTISSA HA PIU' DI 5 ELEMENTI, SI OPERANO DUE METODI:

1) TRONCAMENTO: SI TAGLIANO SEMPLICEMENTE LE CIFRE IN ECCESSO $\rightarrow \tilde{a} = 0.34567 \cdot 10^0$

2) ARROTONDAMENTO: SI APPLICA LA REGOLA DI ARROTONDAMENTO BASATO SULLA CIFRA SUCCESSIVA. E' PREFERITO PIU' DEL TRONCAMENTO PERCHE' E' PIU' ACCURATO; IL TRONCAMENTO INTRODUCE SEMPRE UN ERRORE SISTEMATICO VERSO IL BASSO. $\tilde{a} = 0.34568 \cdot 10^0$

$$d = 0.345678 \cdot 10^{31} \quad (\text{ESPOENTE NON IN RANGE})$$

DATO $d = 3.45678 \cdot 10^{49} \in \mathbb{R}$ PASSARE AL SUO RAPPRESENTANTE \tilde{d} :

Domanda 1) E' IN NOTAZIONE SCIENTIFICA? (SI)

Domanda 2) L'ESPOENTE E' NEL RANGE? (NO) \rightarrow Over-Flow

Over-Flow: NUMERO NON MEMORIZZABILE, QUINDI ANCHE NON RAPPRESENTABILE; TIPICAMENTE RESTITUISCE $\pm \infty$ o SEGNALA ERRORE.

$$d = 0.345678 \cdot 10^{-51} \quad (\text{ESPOLENTE NON IN RANGE})$$

• DATO $d = 0.00345678 \cdot 10^{-49}$ E IR PASSARE AL SUO RAPPRESENTANTE \tilde{d} :

↳ DOMANDA 1) E' IN NOTAZIONE SCIENTIFICA? (SI)

↳ DOMANDA 2) L'ESPOLENTE E' NEL RANGE? (NO) → Under-Flow

Under-Flow: NUMERO TROPPO PICCOLO PER ESSERE RAPPRESENTATO NORMALMENTE. IL SISTEMA PUO:

RESTITUIRE ZERO, USARE NUMERI DENORMALIZZATI (SE SUPPORTATI), SENZA UN ERRORE DI UNDER FLOW.

DEFINITO I NUMERI FINITI E COME VENGONO MEMORIZZATI IN UN CALCOLATORE, RESTA CAPIRE COME UN NUMERO REALE VENGA APPROSSIMATO DA UN NUMERO FINITO.

DATO UN NUMERO REALE NON NULLO $d = \pm(d_1 d_2 \dots) \cdot \beta^p$ t.c. $1 \leq p \leq w$, $d_i = 0 \forall i > t$ (cioè TUTTI ZERI DOPO LE PRIME t CIFRE = n° AL PIÙ t CIFRE), ALLORA $d \in \mathbb{F}(\beta, t, 1, w)$. SE NON SIAMO IN QUESTA SITUAZIONE $d \notin \mathbb{F}(\beta, t, 1, w)$ E QUINDI BISOGNA ASSOCIAGLI UN NUMERO FINITO \tilde{d} CHE INDICHEREMO CON $fl(d)$ (SI PRONUNCIA FLOAT DI d). SUPPOSTO d POSITIVO (ANALOGO PER QUELLO NEGATIVO) E LA BASE β PARI, SI HANNO I SEGUENTI CASI:

• PER $p \in [\lambda, w]$ VIENE SEGNALATA UNA CONDIZIONE DI ERRORE: $\begin{cases} 1) p < \lambda \rightarrow \text{Under Flow} \\ 2) p > w \rightarrow \text{Over Flow} \end{cases}$

• PER $p \in [\lambda, w]$ MA $d \notin \mathbb{F}(\beta, t, \lambda, w)$ PERCHE' LE SUE CIFRE d_i CON $i > t$ NON SONO TUTTE NULLE: VIENE ASSEGNATO UN NUMERO FINITO $fl(d)$ SEGUENDO DUE POSSIBILI CRITERI:

$$\text{TRONCAMENTO } d \text{ AUA } t\text{-ESIMA CIFRA} \quad fl_T(d) = \pm \left(\sum_{i=1}^t d_i \beta^{i-t} \right) \beta^p$$

ESEMPIO: SIA $d = 1234.36789$ E VOGLIAMO SOLO 2 CIFRE SIGNIFICATIVE ($t=2$)

↳ PASSO 1, NORMALIZZO IL NUMERO: $d = 0.\overbrace{123456789}^{\text{MANTISSA}} \cdot 10^4$ (HO SPOSTATO LA VIRGOLA DI 4 CIFRE, $p=4$)

↳ PASSO 2, PREMO SOLO t CIFRE SIGNIFICATIVE: $0.\overbrace{123456789}^{t=2} \rightarrow 0.12$

↳ PASSO 3, APPLICO LA FORMULA DEL TRONCAMENTO: $fl_T(d) = \left(d_1 \beta^{-2} + d_2 \beta^{-1} \right) \cdot \beta^p = (1 \beta^{-2} + 2 \beta^{-1}) \cdot 10^4 = (0.1 + 0.02) \cdot 10^4 = 0.12 \cdot 10^4 = 1200$ ($t=4$: 1234, $t=6$: 1234.56)

$$\text{ARROTONDAMENTO } d \text{ AUA } t\text{-ESIMA CIFRA} \quad fl_A(d) = \pm fl_T \left(\left(\sum_{i=1}^{t+1} d_i \beta^{-i} + \frac{\beta}{2} \beta^{-t-1} \right) \beta^p \right)$$

QUINDI:

• SE $d_{t+1} < \frac{\beta}{2}$ ALLORA SI HA $fl_A(d) = fl_T(d)$ PASSO 4 } PASSO 3

• SE INVECE $d_{t+1} \geq \frac{\beta}{2}$, ALLORA SI HA $fl_A(d) = fl_T(d) + \beta^{p-t}$ } PASSO 5

ESEMPIO: SIA $d = 1234.36789$ E VOGLIAMO SOLO 4 CIFRE SIGNIFICATIVE ($t=4$)

↳ PASSO 1, NORMALIZZO IL NUMERO: $d = 0.\overbrace{123456789}^{\text{MANTISSA}} \cdot 10^4$ (HO SPOSTATO LA VIRGOLA DI 4 CIFRE, $p=4$)

↳ PASSO 2, IDENTIFICA LE CIFRE: $d_1=1$; $d_2=2$; $d_3=3$; $d_4=4$; $d_5=5 \leftarrow$ QUESTA E' $d_{(t+1)}$. UNA CIFRA CHE DECIDE)

↳ PASSO 3, CONFRONTA d_5 CON $\frac{\beta}{2}$: $d_5=5$; $\frac{\beta}{2} = \frac{10}{2} = 5$; $5 \geq 5$? (SI, VERO)

↳ PASSO 4, CALCOLARE IL TRONCAMENTO (BASE DI PARTENZA): $fl_T(d) = 0.1234 \cdot 10^4 = 1234$

↳ PASSO 5, AGGIUNGI L'INCREMENTO β^{p-t} : $\beta^{(p-t)} = 10^{4-4} = 10^0 = 1$;

$$fl_A(d) = fl_T(d) + \beta^{p-t} \rightarrow fl_A(d) = fl_T(d) + 1 \rightarrow fl_A(d) = 1234 + 1 \rightarrow fl_A(d) = 1235$$

OSSERVAZIONE (IL SANDWICH): SIANO x, y DUE NUMERI FINITI CONSECUTIVI TALI CHE $x \leq d < y$. ALLORA

$$x = \left(\sum_{i=1}^t d_i \beta^{-i} \right) \beta^p, \quad y = \left(\sum_{i=1}^t d_i \beta^{-i} + \beta^{-t} \right) \beta^p \quad \text{E RISULTA:}$$

$$fl_T(d) = x \quad ; \quad fl_A(d) = \begin{cases} x & \text{SE } d < \frac{x+y}{2} \\ y & \text{SE } d \geq \frac{x+y}{2} \end{cases}$$

ESEMPIO NUMERICO: SIA $x = 3.141$ E $y = 3.142$ ILN PUNTO MEDIO QUINDI $\frac{(3.141 + 3.142)}{2} = 3.1415$.

SE $d = 3.14149$: $d < 3.1415 \rightarrow fl_A(d) = 3.141$ (x)

SE $d = 3.14159$: $d \geq 3.1415 \rightarrow fl_A(d) = 3.142$ (y)

Ogni numero finito rappresenta se stesso e un intero intervallo di numeri reali: quando usiamo numeri in virgola mobile con t cifre, ogni numero rappresentabile "cattura" tutti i numeri reali in un certo intervallo ad esso.

TRONCAMENTO: SEMPRE PER DIFETTO (PIÙ PESSIMISTICO)

ARROTONDAMENTO: VERSO IL PIÙ VICINO (PIÙ ACCURATO)

ERROTI DI RAPPRESENTAZIONE:

Esempio 1.2 Definito l'insieme dei numeri finiti $\mathbb{F}(10, 5, -50, 49)$ resta definito il numero di posizioni (bit nel caso di base 2) necessarie per rappresentare in memoria un numero finito dell'insieme. Nel caso specifico saranno 1 per il segno (0 se positivo e 1 se negativo), 2 per l'esponente (si usa la tecnica di memorizzazione per traslazione, cioè nei due campi per l'esponente si memorizzano i valori da 00 a 99 intendendo gli esponenti da -50 a 49) e 5 posizioni per la mantissa (si memorizza a partire da sinistra). Vediamo qualche esempio numerico:

$$\alpha = 0.1039 \times 10^{-6} \quad 04410390$$

dove 0 indica che il numero è positivo, 44 rappresenta l'esponente -06, quindi la mantissa 10390 arrotondata alla quinta cifra.

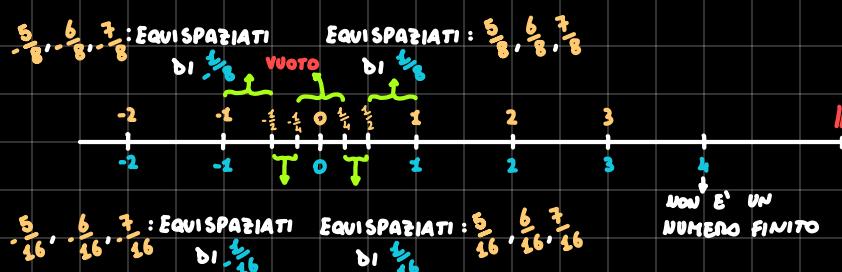
$\alpha = 0.05302$	04953020
$\alpha = -237.141$	95323714
$\alpha = -0.00321665$	94832167

RISULTA EVIDENTE CHE L'INSIEME DEI NUMERI FINITI RAPPRESENTA SOLO UN RISTRETTO SOTTOINSIEME DI QUELLO DEI NUMERI REALI. LA MAGGIORPARTE DEI VALORI $d \in \mathbb{R}$ RISULTA $\notin \mathbb{F}(\beta, t, \lambda, w)$, quindi tali valori POSSANO ESSERE SOLAMENTE APPROSSIMATI MEDIANTE UN $\tilde{d} \in \mathbb{F}(\beta, t, \lambda, w)$, COMMETTENDO UN CERTO ERRORE DI RAPPRESENTAZIONE.

Per valutare l'entità si definiscono le seguenti quantità:

$$E_{\text{ABS}} = |\tilde{d} - d| \quad \text{ERRORE ASSOLUTO} \quad \text{e} \quad E_{\text{REL}} = \left| \frac{\tilde{d} - d}{d} \right| \quad \text{SE } d \neq 0 \quad \text{ERRORE RELATIVO}$$

La rappresentazione discreta della retta reale descritta in precedenza:



È TALE CHE FORNISCE UN ERRORE RELATIVO DI RAPPRESENTAZIONE MASSIMO COSTANTE PER OGNI d , MENTRE QUELLO ASSOLUTO, DI CONSEGUENZA, AUMENTA PROPORTIONALMENTE AL VALORE $|d|$.

SULLA RETTA REALE (CONTINUA), I NUMERI RAPPRESENTABILI SONO PUNTI DISCRETI (ISOLETTI). Lo spazio tra questi punti (il buco sulla retta) cresce proporzionalmente al valore $|d|$:

- INTORNO A 1: |---| (BUCHI PICCOLI = 0.1)
 - INTORNO A 100: |---|---| (BUCHI GRANDI = 10)
 - INTORNO A 10000: |-----|----| (BUCHI ENORMI = 10000)
- } ERRORE MASSIMO POSSIBILE = META' DELLA DISTANZA TRA DUE N° CONSECUTIVI
 } DISTANZA CRESCE = ERRORE ASSOLUTO CRESCE
 } HA LA DISTANZA E' SEMPRE UNA FRAZIONE COSTANTE DI β = ERRORE RELATIVO COSTANTE

ESEMPIO: Sia $d = 1.234567$, i numeri rappresentabili vicini sono $\underline{1.2}$ - $\underline{1.234567}$ - $\underline{1.2345671}$

↳ Arrotondo a $\tilde{\alpha} = 1.2$
 $E_{\text{ass}} = |1.234567 - 1.2| = 0.034567$ $E_{\text{rel}} = \frac{|1.234567 - 1.2|}{1.234567} = 2.8\%$

ESEMPIO: Sia $d = 12345.67$, i numeri rappresentabili vicini sono $\underline{12000}$ - $\underline{12345.67}$ - $\underline{12345.671}$

↳ Arrotondo a $\tilde{\alpha} = 12000$
 $E_{\text{ass}} = |12345.67 - 12000| = 345.67$ $E_{\text{rel}} = \frac{|12345.67 - 12000|}{12345.67} = 2.8\%$

NOTA CHE t E' FONDAMENTALE, ESSO DETERMINA LA DENSITA' DEI PUNTI SULLA RETTA, SEDE LA RELAZIONE:
 "t PIU' GRANDE = PIU' NUMERI RAPPRESENTABILI = BUCHI PIU' PICCOLI = MAGGIORE PRECISIONE"; t DETERMINA LA
 RISOLUZIONE / PRECISIONE DELLA TUA RAPPRESENTAZIONE SULLA RETTA.

NEL CALCOLO SCIENTIFICO, DOVE LA RISPOSTA AI PROBLEMI POSSONO VARIARE GRANDEMENTE IN VALORE, SOLITAMENTE SI USA L'ERRORE RELATIVO IN QUANTO E' SAVING VARIANT (L'ERRORE RELATIVO NON CAMBIA SE MOLTIPLICHI IL NUMERO PER UNA COSTANTE s). INFATTI PER $d \rightarrow sd$ E $\tilde{\alpha} \rightarrow s\tilde{\alpha}$, E_{rel} RESTA USUALE.
 QUESTO RENDE L'ERRORE RELATIVO LA MISURA GIUSTA PER IL CALCOLO SCIENTIFICO, PERCHE' E' INDEPENDENTE DALL'ORDINE DI GRANDEZZA DEI NUMERI (AL CONTRARIO DELL'ERRORE ASSOLUTO).

Th. LIMITI DEGLI ERRORI ASSOLUTI

PER OGNI $d \in \mathbb{R}$ RISULTA: $|fl_T(d) - d| < \beta^{p-t}$, $|fl_A(d) - d| \leq \frac{1}{2}\beta^{p-t}$ DOVE IL SEGNO DI UGUALIANZA VALE SOLO SE $d_{t+1} = \beta/2$ E $d_{t+i} = 0 \quad i \geq 2$.
 SULLA RETTA β^{p-t} INDICA LA DISTANZA TRA DUE NUMERI FINITI CONSECUTIVI, $x - y = \beta^{p-t}$

TRONCAMENTO: ERRORE $< \beta^{p-t}$

↳ TRONCHI SEMPRE PER DIFETTO

↳ L'ERRORE PUO' ESSERE QUASI QUANTO LA DISTANZA TRA DUE NUMERI CONSECUTIVI (MA MAI UGUALE)

↳ NEL CASO PISSIORE PERDI QUASI UNA CIFRA INTERA

ARROTONDAMENTO: ERRORE $\leq \frac{1}{2}\beta^{p-t}$

↳ ARROTONDI AL PIU' VICINO

↳ L'ERRORE E' MASSIMO META' DELLA DISTANZA TRA DUE NUMERI CONSECUTIVI

↳ DUE VOLTE PIU' PRECISO DEL TRONCAMENTO

ESEMPIO NUMERICO ($\beta=10, t=2$): SUPPONIAMO DI AVERE $d = 1.299 \rightarrow d = 0.1299 \cdot 10^3$ CON $p = 1$; DISTANZA TRA NUMERI CONSECUTIVI: $\beta^{p-t} = 10^{-2} = 0.1$

↳ CON TRONCAMENTO: $x = 1.2$, $y = 1.3$ $fl_T(1.299) = 1.2$

↳ ERRORE = $|1.299 - 1.2| = 0.099$, LIMITE TEOREMA: < 0.1 (RISPETTATO)

↳ CON ARROTONDAMENTO: $fl_A(1.299) = 1.3$ (PIU' VICINO)

↳ ERRORE = $|1.299 - 1.3| = 0.001$, LIMITE TEOREMA: ≤ 0.05 (RISPETTATO, MOLTO MEGLIO)

$|fl_A(d) - d| \leq \frac{1}{2}\beta^{p-t}$ { L'UGUALIANZA VALE SOLO SE $d_{t+1} = \beta/2$ (ESATTAMENTE SUL PUNTO MEDIO) E $d_{t+i} = 0$ PER $i \geq 2$ (TUTTE LE ALTRE CIFRE SONO ZERO)}

ESEMPIO : $d = \frac{x+y}{2}$ ESATTAMENTE, $x = 1.2$, $y = 1.3$, $d = 1.25$ (PUNTO MEDIO PERFETTO)
 $fl_A(d)$ PUO' ESSERE $\frac{1}{2}\beta^{p-t}$ (UQUASIANZA). SE $d = 1.251$ (NON PIU' SUL PUNTO MEDIO, ERRORE = $0.049 < 0.05$ (DISUQUASIANZA STRETTA))

Dim. Siano x e y i due numeri consecutivi tali che $x \leq d < y$ RIPRENDENDO L'OSSERVAZIONE:
OSSERVAZIONE (IL SANDWICH): Siano x, y due numeri finiti consecutivi tali che $x \leq d < y$ Allora

$$x = \left(\sum_{i=1}^t d_i \beta^{-i} \right) \beta^p, \quad y = \left(\sum_{i=1}^t d_i \beta^{-i} + \beta^{-t} \right) \beta^p \quad \text{e risulta:}$$

$$fl_T(d) = x; \quad fl_A(d) = \begin{cases} x & \text{se } d < \frac{x+y}{2} \\ y & \text{se } d \geq \frac{x+y}{2} \end{cases}$$


TRONCAMENTO E' SEMPRE x

$$\text{PER } fl_T(d) = x; \quad fl_A(d) = \begin{cases} x & \text{se } d < \frac{x+y}{2} \\ y & \text{se } d \geq \frac{x+y}{2} \end{cases}$$

SARÀ: $d - fl_T(d) < \underbrace{y-x}_{\text{"DISTANZA TRA DUE NUMERI CONSECUTIVI}} = \beta^{p-t};$
 $d - x < y - x = \beta^{p-t}$

$d < y$ (PER IPOTESI: d E' TRA x E y), QUINDI $d - x < y - x = \beta^{p-t}$, CONCLUSIONE:
 $|fl_T(d) - d| = d - fl_T(d) = d - x < \beta^{p-t}.$

Ancora SARÀ: $|fl_A(d) - d| \leq \frac{y-x}{2} = \frac{1}{2}\beta^{p-t}$ E L'UQUASIANZA VALE SOLO SE $d = \frac{x+y}{2}$,
cioè $d_{t+2} = \beta/2$ E $d_{t+i} = 0$ PER $i \geq 2$.

DEFINIZIONE: DATO L'INSIEME DI n FINITI $\mathbb{F}(\beta, t, \lambda, w)$, SI DICE UNITA' DI ARROTONDAMENTO E LA SI INDICA CON u , LA QUANTITA':

$$u = \begin{cases} \beta^{t-t} & \text{PER TRONCAMENTO} \\ \frac{1}{2}\beta^{t-t} & \text{PER ARROTONDAMENTO} \end{cases}$$

L'UNITA' DI ARROTONDAMENTO E' UNA MISURA DELLA PRECISIONE MASSIMA DEL SISTEMA FLOATING-POINT.

E' L'ERRORE RELATIVO MASSIMO CHE PUOI COMMETTERE RAPPRESENTANDO UN NUMERO.

PERCHE' β^{t-t} ? LA DEFINIZIONE DI u CERCA IL CASO PESSIMO POSSIBILE. L'ERRORE RELATIVO PER UN NUMERO d SPECIFICO E': ERRORE RELATIVO = $\frac{\text{ERRORE ASSOLUTO}}{|d|}$, SAPPIAMO CHE $|fl(d) - d| \leq \beta^{p-t}$ (PER ARROTONDAMENTO CON $\frac{1}{2}\beta^{t-t}$) DOVE p E' L'ESponente DI d . VOGLIAMO trovare IL MASSIMO POSSIBILE DI E_{REL} :

$E_{REL} = \frac{\beta^{p-t}}{|d|}$, PER MASSIMIZZARE QUESTA FRAZIONE: NUMERATORE β^{p-t} E' FISSO (DATO DA p), MENTRE IL DENOMINATORE $|d|$ DEVE ESSERE MINIMO. QUALE' IL MINIMO $|d|$ CON ESPOENTE p :

Per un numero normalizzato con esponente p : $d = (0.d_1 d_2 d_3 \dots) \cdot \beta^p$ DOVE $d_1 \geq 1$. IL MINIMO E'
QUANDO $d_1 = 1$ E TUTTE LE ALTRE CIFRE SONO 0: $|d|_{MIN} = 0.1000\dots \cdot \beta^p = \beta^{-1} \cdot \beta^p = \beta^{p-1}$

Quindi il caso pessimo: $E_{REL_{MAX}} = \frac{\beta^{p-t}}{\beta^{p-1}} = \beta^{((p-t)-(p-1))} = \beta^{(p-t-p+1)} = \beta^{(t-1)}$

Th. $\forall d \in \mathbb{R}$ e $d \neq 0$ vale:

$$\left| \frac{\delta f(d) - d}{d} \right| < u$$

(QUANTO SBAGLIAMO RISPETTO AL VALORE ESATTO)

$$u = \begin{cases} \beta^{t-t} \text{ PER TRONCAMENTO} \\ \frac{1}{2} \beta^{t-t} \text{ PER ARROTONDAMENTO} \end{cases}$$

Dim. Caso di troncamento:

FORMA PIÙ SEMPLICE

$$|d| = (d_1 \beta^{-1} + d_2 \beta^{-2} + \dots) \beta^p \geq d_1 \beta^{-1} \beta^p \geq \beta^{p-1}$$

$$\left| \frac{d - \delta f(d)}{d} \right| < \frac{\beta^{t-t}}{\beta^{p-1}} = \beta^{t-p}$$

SAPPIAMO CHE $|d| \geq \beta^{p-1}$, QUINDI $\frac{1}{d} \leq \frac{1}{\beta^{p-1}}$ E' IL LIMITE SUPERIORE. PASSIAMO DAL LIMITE INFERIORE AL LIMITE SUPERIORE.

NON POSSO CALCOLARE L'ERRORE ESATTO OGNI VOLTA, SAREBBERE TROPPO COSTOSO. VOLGIO SAPERE "QUAL'È IL MASSIMO ERRORE CHE POSSO ASPETTARMI?"

IL TEOREMA DICE: "QUALUNQUE SIA IL NUMERO CHE RAPPRESENTA IN FLOATING-POINT, L'ERRORE RELATIVO SARÀ SEMPRE MINORE DI $u = \beta^{t-t}$ ".

IN SINTESI, MIGLIORARE L'ERRORE VUOL DIRE TROVARE UNA GARANZIA SUL MASSIMO ERRORE POSSIBILE. È COME DIRE "NEL PEZZOPIRE DEI CASI, SBASLI AL MASSIMO DI QUESTA QUANTITÀ". QUESTO CI DA' SICUREZZA NEI CALCOLI NUMERICI.

BASE DEL SISTEMA NUMERICO

$$\left| \frac{d - \delta f(d)}{d} \right| < \frac{\beta^{t-t}}{\beta^{p-1}} = \beta^{t-p}$$

$p = \text{POSIZIONE DELL'ESPONENTE}$ $t = \text{NUMERO DI CIFRE DI MANTISSA}$

L'ERRORE RELATIVO È LIMITATO SUPERIORMENTE DA β^{t-p} , O ANCORÀ PIÙ SEMPLICEMENTE, È MINORE DI β^{t-t} .

- 1) PRENDIAMO UN N° REALE d 2) LO RAPPRESENTAMO IN $\delta f(d)$ 3) CALCOLIAMO E_{REL} 4) Th. dice che E_{REL} È LIMITATO A β^{t-t}

Dim. Caso di arrotondamento:

$$\left| \frac{d - \delta f(d)}{d} \right| < \frac{1}{2} \frac{\beta^{t-t}}{\beta^{p-1}} = \frac{1}{2} \beta^{t-p}$$

"QUALUNQUE SIA IL NUMERO CHE RAPPRESENTA IN FLOATING-POINT, L'ERRORE RELATIVO SARÀ SEMPRE MINORE DI $u = \frac{1}{2} \beta^{t-p}$ ".

NELLA CATENA DI MAGGIORANZA: $|d| = (d_1 \beta^{-1} + d_2 \beta^{-2} + \dots) \beta^p \geq d_1 \beta^{-1} \beta^p \geq \beta^{p-1}$, L'USCILLIANZA SI POTREBBE AVERE S.S.S. $d_{t+2} = \beta/2$ E $d_{t+i} = 0 \quad \forall i \geq 2$, MA IN TAL CASO CORRISPONDEREBBE:

$d \geq (d_1 \beta^{-1} + d_2 \beta^{-2} + \dots) \beta^p > d_1 \beta^{-1} \beta^p \geq \beta^{p-1}$, QUESTO DICE CHE IL LIMITE INFERIORE È "ROBUSTO" ANCHE SENZA IL VALORE ASSOLUTO.

L'ERRORE RELATIVO CHE SI COMMETTE NEL RAPPRESENTARE IL NUMERO REALE d CON UN NUMERO FINITO $\delta f(d)$ LO INDICHIAMO CON ϵ E LO CHIAMEREMO "ERRORE RELATIVO DI RAPPRESENTAZIONE":

$$\epsilon = \frac{\delta f(d) - d}{d}$$

IL TH. DI PRIMA DICE CHE L'UNITÀ DI ARROTONDAMENTO LIMITA SUPERIORMENTE IL MODULO DEL' ERRORE RELATIVO DI RAPPRESENTAZIONE.

→ MODO ALTERNATIVO E ELEGANTE DI ESPRIMERE L'ERRORE DI RAPPRESENTAZIONE IN FLOATING - POINT !

* (COROLARIO : $\forall d \in \mathbb{R}, d \neq 0$ VALE $f_l(d) = d(1 + \epsilon)$, CON $|\epsilon| < u$ ($f_l(d) \approx d$ CON PERTURBAZIONE RELATIVA PICCOLA))

DIM.

$$1) \left| \frac{f_l(d) - d}{d} \right| < u \Rightarrow \left| \frac{f_l(d)}{d} - \frac{d}{d} \right| < u \Rightarrow \left| \frac{f_l(d)}{d} - 1 \right| < u$$

$$\left. \begin{array}{l} * \quad \epsilon = \frac{f_l(d) - d}{d} \\ \text{ERRORE RELATIVO} \\ \text{RISPETTO AD } d \end{array} \right\}$$

$$2) \epsilon = \frac{f_l(d)}{d} - 1 \quad * \quad |\epsilon| < u \Rightarrow \frac{f_l(d)}{d} = 1 + \epsilon \Rightarrow \frac{f_l(d)}{d} = \frac{d(1 + \epsilon)}{d} = d(1 + \epsilon) \Rightarrow f_l(d) = d(1 + \epsilon)$$

MOLTIPLICO PER d

ANALOGAMENTE AL Th. DI PRIMA, SE $f_l(d) \neq 0$ VALE
DEFINENDO $\epsilon = \frac{d - f_l(d)}{f_l(d)}$ VALE $f_l(d) = \frac{d}{1 + \epsilon}$ CON $|\epsilon| < u$.

$$\left| \frac{f_l(d) - d}{f_l(d)} \right| < u, \text{ DA QUESTO SI HA Poi CHE}$$

$$\left. \begin{array}{l} * \quad \epsilon = \frac{f_l(d) - d}{f_l(d)} \\ \text{ERRORE RELATIVO} \\ \text{RISPETTO AD } f_l(d) \end{array} \right\}$$

ESEMPIO: PER $f_l(d) = d(1 + \epsilon)$, SE VOLLO SOMMARE

DUE NUMERI x E y CON IL COMPUTER:

$$\left. \begin{array}{l} 1) \text{PREndo } x \text{ ESATTO E LO RAPPRESENTO: } f_l(x) = x(1 + \epsilon_1) \\ 2) \text{PREndo } y \text{ ESATTO E LO RAPPRESENTO: } f_l(y) = y(1 + \epsilon_2) \\ 3) \text{FAI LA SOMMA IN FLOATING-POINT: } f_l(x+y) = (x+y)(1 + \epsilon_3) \end{array} \right\} \begin{array}{l} \text{PARTO DAI VALORI ESATTI E MOLTIPLICO PER } (1 + \epsilon) \\ \text{PER OTTENERE LA RAPPRESENTAZIONE.} \end{array}$$

ESEMPIO: PER $f_l(d) = \frac{d}{1 + \epsilon}$, SUPPONIAMO DI AVERE GIÀ CALCOLATO DAL COMPUTER E VOGLI CAPIRE QUAL'ERA IL VALORE ESATTO:

$$\left. \begin{array}{l} 1) \text{IL COMPUTER TI DA } f_l(d) = 3.14159 \\ 2) \text{TU SAI CHE } f_l(d) = \frac{d}{1 + \epsilon} \text{ CON } |\epsilon| < 10^{-6} \\ 3) \text{QUINDI } d = f_l(d) \cdot (1 + \epsilon) \approx 3.14159 \cdot (\text{QUALcosa DI VICINO A } 1) \end{array} \right\} \begin{array}{l} \text{PARTO DAL VALORE RAPPRESENTATO } f_l(d) \text{ E LO} \\ \text{USO PER STIMARE L'ORIGINALE } d. \end{array}$$

- $d(1 + \epsilon) =$ VAI DA ESATTO → RAPPRESENTATO (ANALISI SIRETTA)
- $\frac{d}{(1 + \epsilon)} =$ VAI DA RAPPRESENTATO → ESATTO (ANALISI INVERSA)

ARITMETICA FLOATING-POINT: OLTRE ALL'ERRORE INTRODOTTO NELLA RAPPRESENTAZIONE DI UN NUMERO REALE, ANCHE LE SINGOLE OPERAZIONI RISULTANO APPROSSIMATE. Ad ESEMPIO, LA SOMMA DI DUE NUMERI FINITI x E y CHE $\epsilon \in \mathbb{F}(p, t, l, w)$ E' UN NUMERO CHE PUO' NON APPARTENERE ALL'INSIEME $F(p, t, l, w)$.

ES: SIANO $0.1 \cdot 10^{-3}$ E $0.1 \cdot 10^3$ E $\mathbb{F}(10, 3, 2, w)$. ESEGUENDO LA SOMMA SI HA:

$$\left. \begin{array}{l} 100.0000 + 0.0001 = 0.1000001 \cdot 10^3 \in \mathbb{F} \text{ PERCHE' LA SUA RAPPRESENTAZIONE ESATTA RICHIEDE} \\ 0.0001 = 7 \text{ CIFRE PER LA MANTISSA. QUESTO PRESENTA IL PROBLEMA DI APPROSSIMARE IL RISULTATO} \\ 100.0001 \text{ DI UN'OPERAZIONE ARITMETICA FRA DUE NUMERI FINITI CON UN NUMERO FINITO.} \end{array} \right\}$$

OCCORRE DEFINIRE QUINDI UN'ARITMETICA DI MACCHINA. Per CONVENIENZA SI ASSUME CHE: SE \tilde{o} E' L'OPERAZIONE DI MACCHINA CHE APPROSSIMA L'OPERAZIONE ESATTA op, PER TUTTI I NUMERI FINITI x, y PER CUI L'OPERAZIONE NON DIA LUOGO A CONDIZIONI DI OVERFLOW OPPURE UNDERFLOW, SIA: $x \tilde{o} y = f_l(x op y)$.

QUESTA CONVENZIONE E IL Th. DI PRIMA, COMPORTANO CHE, SE $x op y \neq 0$, VALE: (ERRORE RELATIVO OPERAZIONE MACCHINA)

→ OPERAZIONE MACCHINA

$$\left| \frac{x \tilde{o} y - (x op y)}{x op y} \right| < u$$

1)

1) NOTA: IL COMPUTER QUANDO ESEGUE UN'OPERAZIONE, PRIMA CALCOLA IL

1) RISULTATO ESATTO x op y, Poi LO RAPPRESENTA IN f_l(x op y)

Ogni singola operazione introduce un errore relativo minore di u . Questo è il "MATTONO FONDAMENTALE" per capire come gli errori si propagano in un algoritmo complesso (errore massimo di u per ogni operazione).

ERRORE RELATIVO DELL' OPERAZIONE MACCHINA : $E = \frac{x \tilde{op} y - (x op y)}{x op y}$

Per il corollario possiamo scrivere : $x \tilde{op} y = (x op y)(1+E)$, $|E| < u$ con E l'errore relativo commesso nell'operazione. Il risultato di un'operazione macchina deve essere piccolo fattore introdotto per ogni operazione, e uguale all'approssimazione con un numero finito del risultato dell'operazione esatta.

L'UNITÀ DI ARROTONDAMENTO u è detta precisione di macchina nel sistema floating-point.

Questo valore rappresenta un limite fondamentale sulla precisione del sistema: indica il massimo errore relativo che può verificarsi quando rappresentiamo un numero reale in formato floating-point. Con il termine "precisione macchina" si indicano in realtà due aspetti distinti ma collegati :

• PRECISIONE DI RAPPRESENTAZIONE : descritta da Th. di prima, che quantifica l'errore commesso nel rappresentare un singolo numero in memoria; qualunque numero viene approssimato con un errore relativo minore di u .

• PRECISIONE DI CALCOLO : riguarda l'errore introdotto dalle operazioni aritmetiche; ogni operazione di macchina introduce un ulteriore errore relativo limitato da u . L'unità di arrotondamento u è la "risoluzione minima" del sistema: è impossibile distinguere due numeri che differiscono per meno di u in termini relativi. Es: $u = 10^{-6}$, il sistema non può distinguere 1.000000 da 1.000001 (differenza relativa uguale ad u)

L'UNITÀ DI ARROTONDAMENTO u è il più piccolo numero finito positivo t.c.: $u + 1 = fl(u+1) > 1$. Questo implica che ogni numero finito $v < u$ sarà $v + 1 = fl(v+1) = 1$.

Affiancando una breve nota sul modulo

Vai da ESEMPIO DECIMALI

Parti da pagina 8



FINISCI TUTTO



SISTEMA IN BIZZA LA
CORREZIONE DELL'ES.
X CASA



Poi passa
agli errori

DIRETTO REALE

DIRETTO COMMERCIALE