Aritmetika pohyblivej rádovej čiarky
(FPU – Float Point Unit)
(FPA -Floating Point Arithmetics)

Presnosť  Rozsah

Opakovanie:

- Počítač je stroj na spracovanie čísiel – číslic
- Poznáme:
  - Prirodzené čísla: 1,2,3,4, ...
  - Celé čísla: -3,-2-1,0,1,2,3,4, ...
  - Racionálne čísla (cč/cč, okrem cč/0) môžeme zapísať v tvare:
    Konečných, resp. nekonečných periodický desatinných zlomkov
  - Iracionálne čísla: zapísané v tvare nekonečných neperiodických desatinných zlomkov
- Čo môžeme zobraziť do N bitov?
  - Celé číslo bez znamienka:

    0 až $2^N - 1$

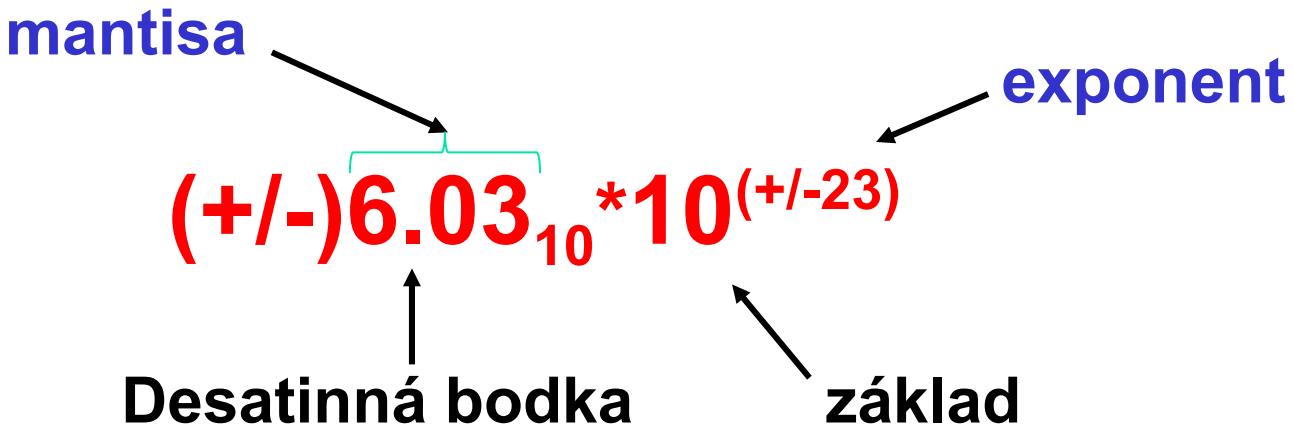
  - Celé číslo so znamienkom (Two's Complement)

    $-2^{(N-1)}$ až $2^{(N-1)} - 1$

A čo iné čísla?

- Veľmi veľké čísla (sekundy v storočí)
  $3{,}155{,}760{,}000_{10}$ ($3.15576_{10} * 10^{9}$)
- Veľmi malé čísla (priemer atómu)
  0.0000000110 ($1.010 * 10^{-8}$)
- Racionálne (periodické) $2/3$ ($0.666666666...$)
- Iracionálne čísla $2^{1/2}$ ($1.414213562373...$)
  e (2.718...), π (3.141...)

Vedecké zobrazenie čísiel (dekadické)



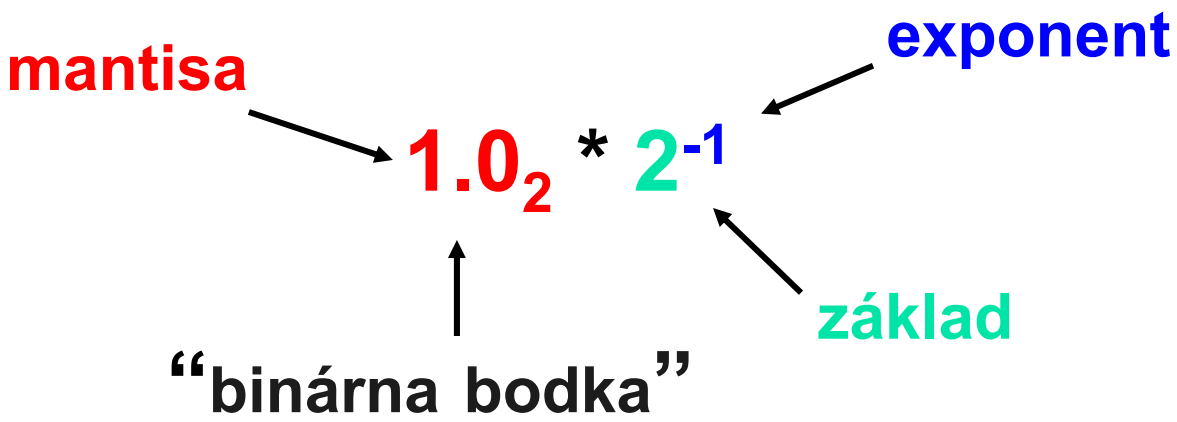
- Normalizovaný tvar zápisu: bez vodiacich núl

(naľavo od desatinnej bodky je len jedna nenulová platná číslica)

- Iný spôsob zápisu: 1/1 000 000 000
  - Normalizovaný: $1.0 * 10^{-9}$
  - Nenormalizovaný: $0.1 * 10^{-8}$

    $10.0 * 10^{-10}$

Vedecké zobrazenie čísiel (binárné)



Potrebujeme zapísať:

- „znamienko“ mantisy
- „znamienko“ exponentu

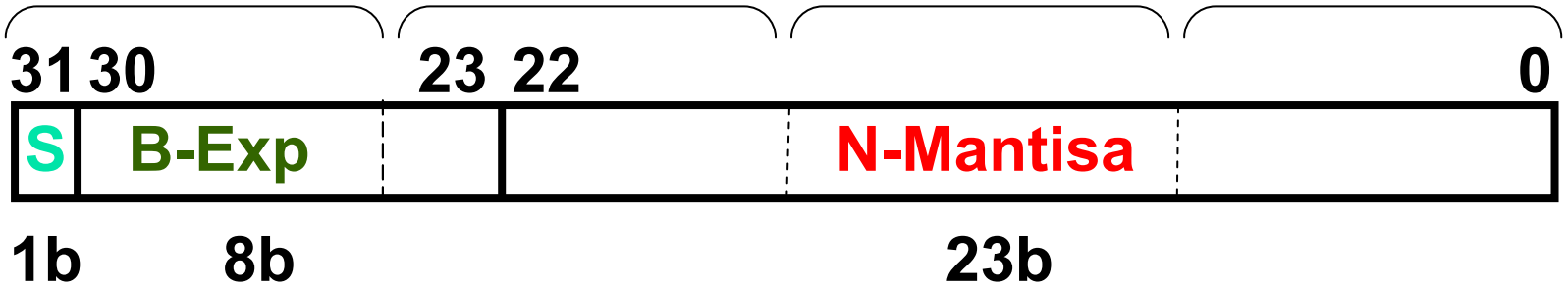
Počítače podporujú prácu s číslami typu float:

Forma zápisu znamienka ???

- priamy kód
- jednotkový doplnok
- dvojkový doplnok

Jednoduchá presnosť čísiel FP (Single Precision – SP, C: float)

- Formát zápisu: (+/-)1.xxxxxxxxxx$_2$*2^{yyyy_2}
- Počet bitov: 32 bits



Mantisa: (priamy kód)

S - Sign znamienko mantisy

|Mantisa| = 1.xxxxxxxxx, xxxxxxxxx = N-Mantisa

- Exponent = B-Exp – Bias, Bias = 127, B-Exp = <1, 254>
- Čísla z rozsahu: $2^{-126}(1.0) \sim 2^{+127}(2 - 2^{-23})$

t.j. $1.18 * 10^{-38} \sim 3.40 * 10^{38}$

Zobrazenie FP čísiel

- Čo sa stane ak je výsledok veľmi veľký?

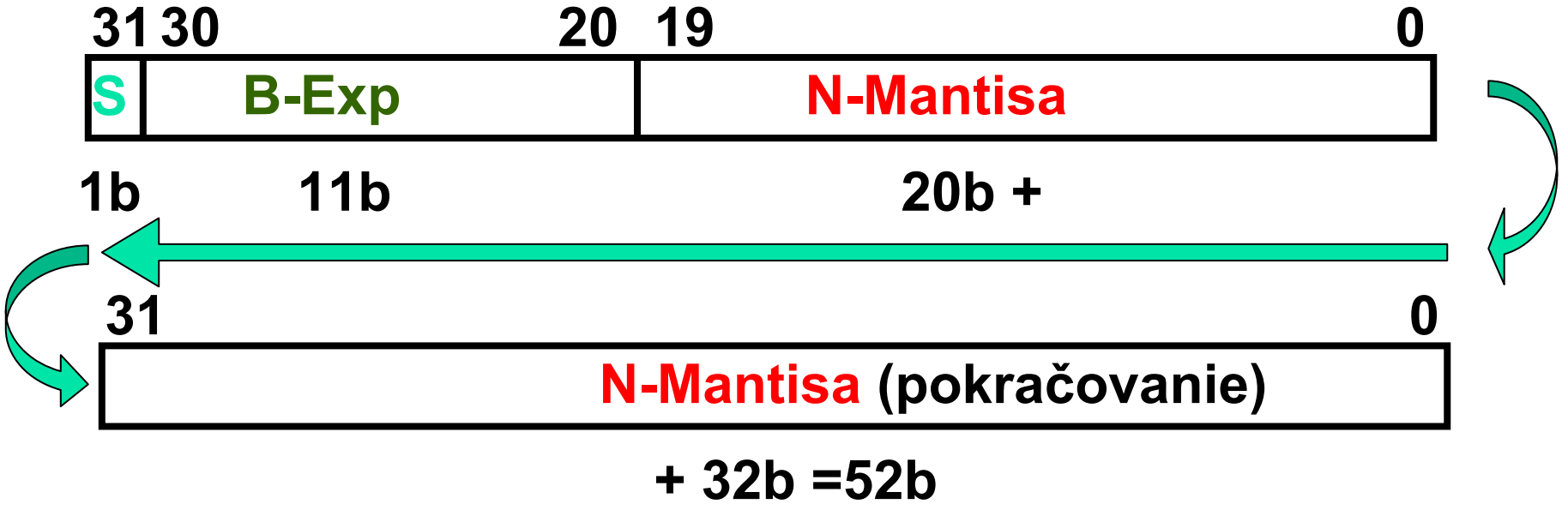
  (> $3.403*10^{38}$) Overflow!

  - Overflow $\Rightarrow$ Exponent väčší ako sa dá zobraziť do 8 bitov
- Čo sa stane ak je výsledok veľmi malý?

  (> 0, < $1.17*10^{-38}$) Underflow!

  - Underflow $\Rightarrow$ Záporný exponent "väčší" ako sa dá zobraziť do 8-bitov
  - Ako zabránime: pretečeniu – overflow,
    podtečeniu – underflow?

Dvojnásobná presnosť čísla FP (Double Precision – DP
C-ko: double 2 * 32 = 64 bitov)



Jednoduchá presnosť ⟷ Dvojnásobná presnosť

Bias: 127 1023 .

Čísla z rozsahu:

$2^{-126}(1.0) \sim 2^{+127}(2-2^{-23})$ $2.0 * 10^{-308} \sim 2.0 * 10^{308}$

Väčšou výhodou je vyššia presnosť

Norma IEEE 754, Zdôvodnenie (1/4)

- Jednoduchá presnosť, Dvojnásobná presnosť
- Znamienkový bit - S: 1 - záporné číslo
  0 - kladné číslo
- Mantisa:
  - Vodiaca jednotka sa nepíše v normovanom čísle
    - $\Rightarrow$ 1 + 23 bitov SP,
    - $\Rightarrow$ 1 + 52 bitov DP
  - Interval:
    - <1, 2) a
    - <0, 1) bez „vodiacej jednotky“
- Poznámka: 0 - číslo nula. Nemá vodiacu – skrytú jednotku,
  $\Rightarrow$ Špeciálny zápis pre vyjadrenie čísla nula,
  rezervovaný špeciálny exponent

Norma IEEE 754 (2/4)

- Niekedy by sme chceli použiť „float“ aj v takom prípade, keď nemáme FP hardware; napr., triediť pomocou celočíselného porovnávania záznamy
- V takomto prípade „rozbijeme“ FP číslo na tri časti
  - Porovnáme znamienka,
  - Porovnáme exponenty,
  - Potom porovnáme normované mantisy
- Dá sa predpokladať, že porovnávanie po skupinách bude rýchlejšie, a zvlášť vtedy, keď porovnávané čísla budú len celé kladné
- Porovnávanie vykonáme v poradí:
  1. Znamienkový bit: záporné < kladné
  2. Exponent: väčšie číslo má väčší exponent
  3. Normovaná mantisa: väčšie číslo má väčšiu mantisu

  Porovnávanie „zastavíme“ pri prvej nezhode

Norma IEEE 754 (3/4)

- Záporný Exponent ?!?!
  - 2's comp? $1.0 * 2^{-1}$?<= >? $1.0 * 2^{+1}$ (1/2 ?<= >? 2)

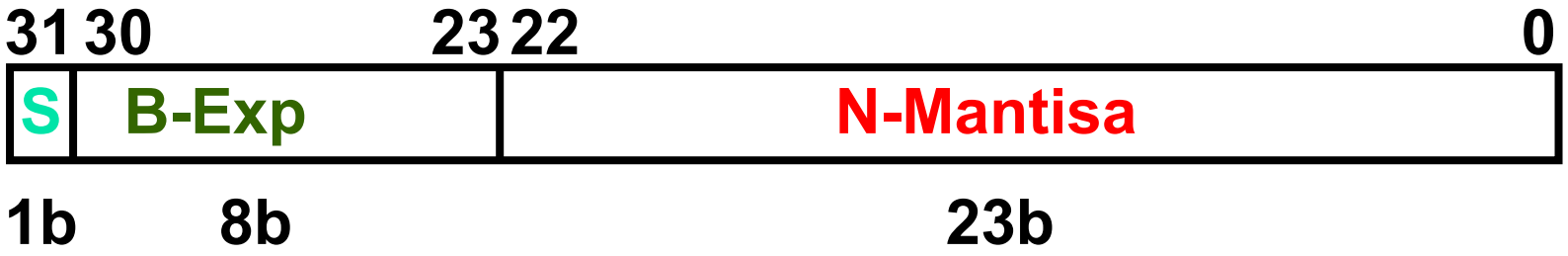
1/2	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

  - „Celočíselné porovnávanie" týchto čísiel
  - Porovnanie: 1/2 ?<= >? 2 dá 1/2 > 2 !
- Celé kladné číslo 0000 0001 je zápornejšie ako, cele kladné číslo 1111 1111 $\Rightarrow$ ľahko sa porovnávajú
  - $1.0 * 2^{-1}$?<= >? $1.0 * 2^{+1}$ (1/2 ?<= >? 2)

1/2	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Norma IEEE 754 (4/4)

- Takéto riešenie sa volá: zápis exponentu s posunutou nulou,
- Ak odpočítame od posunutého exponentu posunutie, dostaneme skutočný exponent
  - IEEE 754: posunutie pre: SP: B = 127
    DP: B = 1023
  - Jednoduchá presnosť:



  - $(-1)^{S} * (1 + \textbf{N-Mantisa}) * 2^{(B\text{-}Exp - 127)}$

Zápis pre DP je rovnaký, len posunutie je 1023 a počet bitov je dvojnásobný

N-Mantisa (1/2)

- Spôsob 1. (Zlomky):
  - Dekadické číslo: $0.340_{10} \Rightarrow \; 340_{10}/1000_{10}$
    $\Rightarrow \; 34_{10}/100_{10}$
  - Binárne číslo: $0.110_2 \Rightarrow \; 110_2/1000_2 = 6_{10}/8_{10}$
    $\Rightarrow 11_2/100_2 = 3_{10}/4_{10}$

N-Mantisa (2/2)

- Spôsob 2. (Hodnota pozície):
  - Dekadicky: $1.6732 = (1*10^{0}) + (6*10^{-1}) + (7*10^{-2}) + (3*10^{-3}) + (2*10^{-4})$
  - Binárne: $1.1001 = (1*2^{0}) + (1*2^{-1}) + (0*2^{-2}) + (0*2^{-3}) + (1*2^{-4})$

***M* = 1.xxx…x_2**

xxx…x**:** bity normovanej mantisy za "binárnou bodkou"

Minimum: 000…0 (*M* = 1.0)

Maximum:111…1 (*M* = 2.0 – ε)

ε- strojová nula

Pr.: Prevod binárneho FP čísla na dekadické

0	0110 1000	101 0101 0100 0011 0100 0010

- Znamienko: 0 $\Rightarrow$ kladné
- Exponent:
  - $0110\ 1000_2 = 104_{10}$ B-Exp
  - **„Vyposúvanie“ exponentu:** 104 - 127 = -23
- Mantisa:
  - $1 + 1*2^{-1} + 0*2^{-2} + 1*2^{-3} + 0*2^{-4} + 1*2^{-5} + \ldots =$

    $= 1 + 2^{-1} + 2^{-3} + 2^{-5} + 2^{-7} + 2^{-9} + 2^{-14} + 2^{-15} + 2^{-17} + 2^{-22} =$

    $= 1.0_{10} + 0.666115_{10}$
- Predstavuje číslo: $1.666115_{10} * 2^{-23} \sim 1.986 * 10^{-7}$

  (približne 2/10 000 000)

Prepočet desatinného čísla na FP číslo (1/3)

- Jednoduché: Ak je menovateľ mocninou 2, t.j. ak (2, 4, 8, 16, atď.), potom je to ľahké.
- Napr.: -0.75
  - -0.75 = -3/4
  - $-11_2/100_2 = -0.11_2$
  - **Normovanie:** $-1.1_2 * 2^{-1}$
  - $(-1)^S * (1 + \textbf{N-Mantisa}) * 2^{(B\text{-}Exp\ -\ 127)}$
  - $(-1)^1 * (1 + .100\ 0000\ ...\ 0000) * 2^{(126\ -\ 127)}$

Prepočet desatinného čísla na FP číslo(2/3)

- Zložitejší prípad: Ak menovateľ nie je mocninou 2.
  - Potom dané číslo nezobrazíme presne.
  - Aby bolo zobrazenie čo najpresnejšie, použijeme „veľa“ bitov mantisy.
  - Keď máme mantisu, správne číslo pre exp. už získame ľahko.
  - ???? Mantisa ??????

Prepočet desatinného čísla na FP číslo (3/3)

- Je zrejmé, že ... Racionálne čísla (x_{10}) majú veľa platných číslic.
- Podobne to platí aj pre ich binárny ekvivalent
- Prepočet racionálneho čísla:

  Ak nevieme zobraziť číslo v tvare $x/ 2^k$

  výsledok prevodu vyzerá nasledovne:

Des. hodnota	**Dvojkové číslo**
	(niekoľko bitov sa zopakuje **)**
1/3	**$0.0101\ 0101\ 01[01]\ldots_2$**
1/5	**$0.0011\ 0011\ 0011\ [0011]\ldots_2$**
1/10	**$0.0001\ 1001\ 1001\ 1[0011]\ldots_2$**

Príklad:

Čo je dekadický ekvivalent FP čísla ?

1	1000 0001	111 0000 0000 0000 0000 0000

1: -1.75
2: -3.5
3: -3.75
4: -7
5: -7.5
6: -15
7: -7 * 2^129
8: -129 * 2^7

Odpoveď:

Dekadický ekvivalent FP čísla:

1	1000 0001	111 0000 0000 0000 0000 0000
S	B-Exp	N-Mantisa

$(-1)^{S} * (1 + \text{N-Mantisa}) * 2^{(\text{B-Exp}-127)}$

$(-1)^{1} * (1 + .111) * 2^{(129-127)}$

$-1 * (1.111) * 2^{(2)}$

-111.1

-7.5

1: -1.75
2: -3.5
3: -3.75
4: -7
5: -7.5
6: -15
7: -7 * 2^129
8: -129 * 2^7

“Na záver ”

- Floating Point čísla sú len náhradou tých čísiel, ktoré sme chceli použiť
- IEEE 754 Floating Point Standard je v praxi najrozšírenejší spôsob zápisu takýchto čísiel
- Od roku ~ 1997 túto normu používa prakticky každý počítač

Viac o FP číslach:

Doteraz sme uvažovali B-Exp v rozsahu:
<1 až 254>

Na čo je použitá „0“ a „255“?

Znázornenie ± ∞

- V FP aritmetike, delenie 0 dá ± ∞, nie pretečenie.
- Prečo?
  - Ak existuje v FP aritmetike ∞ potom výraz X/0 > Y je platné porovnanie
  - IEEE 754 vie zobraziť ± ∞
  - Najkladnejší exponent B-Exp = 255 je rezervovaný pre ∞
  - N-Mantisa je nulová
  - Kladné ∞

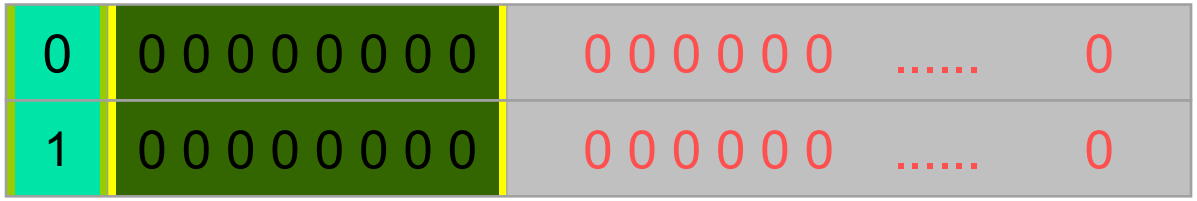
    $+ \infty = +1.0 * 2^{128}$

  - Záporné ∞

    $- \infty = -1.0 * 2^{128}$

Zobrazenie „0“

- Posunutý exponent, samé nuly: B-Exp = 0
- rovnako normovaná mantisa samé nuly
- A čo znamienko?
- +0:
- -0:



- Prečo dve nuly?
  - Výhodné pri limitných porovnávaniach

Špeciálne čísla

- Čo ešte môžeme dodefinovať v (Single Precision) ?

B-Exp	N-Mantisa	Výsledok (číslo)
0	0	0
0	nenulová (!= 0)	???
1-254	nenulová	+/- normované FP čísla
255	0	+/- ∞
255	nenulová (!= 0)	???

- Zostalo nám:
  - (B-Exp = 0) & (N-Mantisa != 0)
  - (B-Exp = 255) & (N-Mantisa != 0)

  … “Skúsime využiť”

Čísla typu: Not a Number (NaN)

- Čo je $\sqrt{-4}$ alebo 0/0?
  - Ak ∞ nie je „chyba“, potom by nemuselo byť ani napr. 0/0 .
  - Zaužíval sa názov: Not a Number (NaN)
  - B-Exp = 255, N-Mantisa nenulová
    - Načo je to dobré?
  - Dá sa predpokladať, že NaN sa využijú pri debagovaní?
  - Napr.: op(NaN, X) = NaN

Zápis nenormovaných čísiel: (1/2)

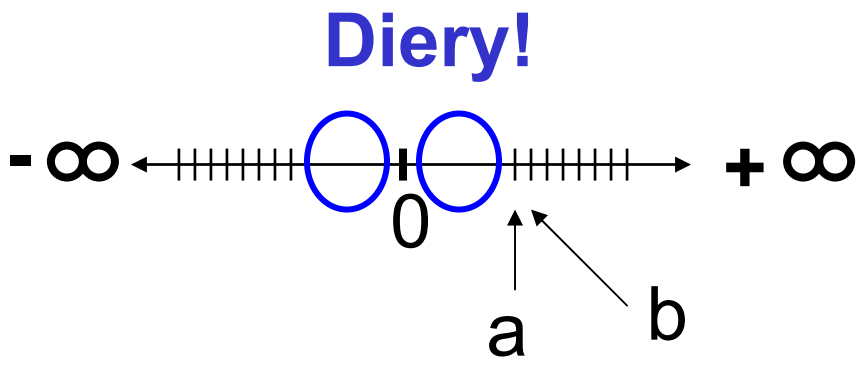
- Problém: FP čísla zapísané v normalizovanom tvare generujú okolo nuly „diery“
  - Najmenšie zobraziteľné kladné číslo:

    $a = 1.0 \ldots 0_2 * 2^{-126} = 2^{-126}$
  - Druhé najmenšie zobraziteľné kladné číslo:

    $b = 1.000\ldots1_2 * 2^{-126} = 2^{-126} + 2^{-149}$

  $a - 0 = 2^{-126}$

  $b - a = 2^{-149}$



Zápis nenormovaných čísiel: s **(2/2)**

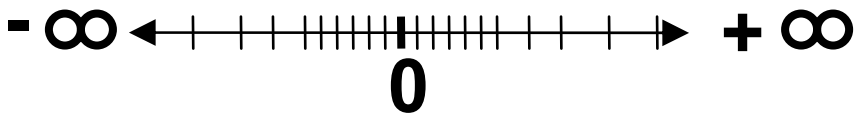
- Riešenie:
  - Zatiaľ sme nepoužili B-Exp = 0, N-Mantisa nenulová
  - Nenormované čísla: bez vodiacej jednotky,

    Najmenší normovaný exponent = -126. (posúvame)
  - Najmenšie zobraziteľné kl. číslo:

    $a = 2^{-149}$ (126+23=149)
  - Druhé najmenšie zobraziteľné kl. číslo:

    $b = 2^{-148}$

    $\Rightarrow$ Diery okolo nuly sú menšie



Zhrnutie

B-Exp	N-Mantisa	Výsledok (číslo)
0	0	0
0	nenulová (!= 0)	Nenormované FP
1-254	nenulová	Normované FP
255	nulová	+/- ∞
255	nenulová (!= 0)	NaN

Zaokrúhlovanie

- Výpočty s reálnymi číslami $\Rightarrow$ problém ako číslo umiestniť do odpovedajúceho priestoru.
- FP hardware obsahuje 2 špeciálne bity pre presnosť

  (zníženie presnosti $\Rightarrow$ zvýšenie rýchlosti)

  00 – 24 bitov (SP)

  10 – 53 bitov (DP)

  11 – 64 bitov (Extended P (vnútorne FPU 80bitov))
- Zaokrúhluje sa vždy pri konvertovaní…
  - DP $\Rightarrow$ SP
  - Číslo FP na integer

IEEE pozná 4 módy zaokrúhlovania:

- Zaokrúhľovanie smerom + ∞
  Vždy nahor : 2.1 ⇒ 3, -2.1 ⇒ -2

- Zaokrúhľovanie smerom - ∞
  Vždy nadol : 1.9 ⇒ 1, -1.9 ⇒ -2

- Odrezanie
  Jednoducho zahoď posledné bity (zaokrúhlenie smerom k 0)

- Zaokrúhlenie na najbližšie číslo (default),
  - vykonáme pripočítaním čísla 1 s váhou o 1 menej, ako je posledný platný rád.
  - resp. párne, ak sú dve najbližšie čísla rovnako vzdialené.

  Pr.: 2.5 ⇒ 2; 3.5 ⇒ 4

Pr.: Zaokrúhlenie na najbližšiu desatinu

2.2499 ≈ 2.2; 2.2501 ≈ 2.3; 2.2500 ≈ 2.2; 2.3500 ≈ 2.4;

Vlastnosti - problémy FP aritmetiky (1/3)

- Presnosť  Rozsah
- Vedecké výpočty vyžadujú chybový menežment
- Nespomenuli sme: Napr.: nie je garantované:
  - $(1/r)*r \neq 1$
  - FPA aritmetika nie je asociatívna !!!!.

  $(A+B) + C \neq A + (B+C)$, $(A*B) * C \neq A * (B*C)$

  Napr: **$(1.0*10^{100} +1.0)- 1.0*10^{100} =0.0$, ale $(1.0*10^{100} - 1.0*10^{100}) +1.0 =1.0$**
  - nie je ani vždy distributívna !!!!

  $(A+B) * C \neq A*C + B*C$
- Implementovanie normy IEEE 754 je ťažké

Vlastnosti - problémy FP aritmetiky (2/3)

- **Súčet a rozdiel**

  Princíp: Majme čísla A=12345 a B=567.89, ktoré sa dajú zapísať v tvare: $A = 1.2345*10^4$, $B = 5.6789*10^2$

Súčet v dekadickom vyjadrení je jednoduchý

$$\begin{array}{r} \mathbf{12345} \\ \mathbf{+\quad 567.89} \\ \hline \mathbf{12912.89} \end{array}$$

ale v FPP aritmetike je treba nanormovať exponenty (zväčšiť menší exponent)

$$\begin{array}{r} \mathbf{1.234500*10^4} \\ \mathbf{+0.056789*10^4} \\ \hline \mathbf{1.291289*10^4} \end{array}$$

Pre binárne čísla je to obdobné, ak treba nanormujeme a zaokrúhlime mantisu výsledku.

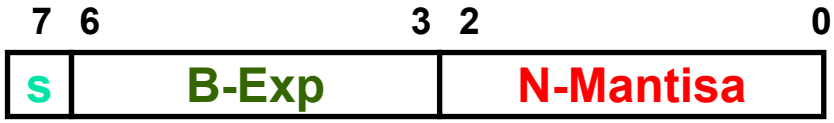
Vlastnosti - problémy FP aritmetiky (3/3)

Zlé implementovanie FPP spôsobuje chyby:

- Viď „Pentium bug“!
- Raketa Patriot – „vojna v zálive“ systém protivzdušnej obrany bol „zapnutý“ cca 100 hodín. Čas generovali ako načítavanie 0,1sek zapísané binárne do 24 b. Binárny ekvivalent 0.1sek je „nekonečné“ číslo => chyba pri zaokrúhlení cca 10^{-7}sek. Za 100hodín chyba narástla na 100 *3600*10*10^{-7} =0,36sek pri rýchlosti 1676m/sek je to cca 600metrov. “Raketa minula prilietavajúci irácky Scud”, ktorý zasiahol tábor americkej armády.
- Ariane 5 (v roku 1996) havarovala kvôli chybe pri konverzii čísla FP (64b) na signed integer (16b) (“nezmestilo sa”) Predpokladaná cena nehody: $500 million

Pr.: Zobrazenie FP čísiel - “malých” (1/3)

- 8-bitov FP číslo
  - Znamienko bit č.7.
  - 4 bity exponent, s posunutím 7.
  - 3 bity normov. mantisa
- Niečo čo sa podobá na IEEE Formát
  - Normované a nenormované čísla
  - zobrazenie 0, NaN, nekonečna)



Pr.: Zobrazenie FP čísiel - “malých” (2/3)

B-Exp	$B\text{-}Exp_2$	E_{10}	2^E	
0	0000	-6	1/64	(nenormované)
1	0001	-6	1/64	(normované)
2	0010	-5	1/32	
3	0011	-4	1/16	
4	0100	-3	1/8	
5	0101	-2	1/4	
6	0110	-1	1/2	
7	0111	0	1	
8	1000	+1	2	
9	1001	+2	4	
10	1010	+3	8	
11	1011	+4	16	
12	1100	+5	32	
13	1101	+6	64	
14	1110	+7	128	
15	1111	+8	(∞, NaN).	

Pr.: Zobrazenie FP čísiel - “malých” (3/3)

	S	B-Exp	N-Man	E_{10}	číslo	
Nenormaliz. čísla	0	0000	000	-6	0	
	0	0000	001	-6	1/8*1/64 = 1/512	← skoro nula
	0	0000	010	-6	2/8*1/64 = 2/512	
	…					
	0	0000	110	-6	6/8*1/64 = 6/512	
	0	0000	111	-6	7/8*1/64 = 7/512	← najväčš.nenorm.
	0	0001	000	-6	8/8*1/64 = 8/512	← najmen. norm.
	0	0001	001	-6	9/8*1/64 = 9/512	
	…					
Normaliz. čísla	0	0110	110	-1	14/8*1/2 = 14/16	
	0	0110	111	-1	15/8*1/2 = 15/16	← skoro 1 (<1)
	0	*0111*	*000*	*0*	*8/8*1 = 1*	
	0	0111	001	0	9/8*1 = 9/8	← skoro 1 (>1)
	0	0111	010	0	10/8*1 = 10/8	
	…					
	0	1110	110	7	14/8*128 = 224	
	0	1110	111	7	15/8*128 = 240	← najväčš. norm
	0	1111	000	8	∞	

Literatúra:

[1] Clements,A: The Principles of Computer Hardware, Oxford

[2] Stalling, W.: Computer Organization and Architecture, principles ...,

[3] Jelšina, M.: Architektúry počítačových systémov, ...