Slankaus kablelio skaičiai

Saulius Gražulis

Vilnius, 2021

Vilniaus universitetas, Matematikos ir informatikos fakultetas
Informatikos institutas

Eksponentinis skaičių užrašymas

$$602\underbrace{00\ldots0}_{\text{21 kartų}} = 6.02 \times 10^{23}$$

$$\pm d_0.d_1d_2\ldots d_{p-1} \times \beta^e = \sum_{i=0}^{p-1} d_i\beta^{-i} \times \beta^e, (0 \leq d_i < \beta)$$

$$\underbrace{602 \times 10^{21}}_{\textit{nenormalizuotas}} = \underbrace{6.02 \times 10^{23}}_{\textit{normalizuotas}} = \underbrace{0.602 \times 10^{24}}_{\textit{nenormalizuotas}}$$

Slankaus kablelio skaičiai

$$\pm d_0.d_1 d_2 \ldots d_{p-1} \times \beta^e = \pm \sum_{i=0}^{p-1} d_i \beta^{-i} \times \beta^e, (0 \leq d_i < \beta)$$

$$\beta = 2$$

$$0.1_{10} \approx +1.10011001100110011001101_2 \times 2^{-4}$$

- (Trupmenos) ženklas
- Laipsnio rodiklis
- Trupmena (mantisė; angl. “significand”)

IEEE 754 standartas

- Dvejetainiai ($\beta = 2$, (IEEE 1985)) ir dešimtainiai ($\beta = 10$, (IEEE 2008)) formatai
- Viengubo, dvigubo tikslumo (IEEE 1985), pusinio, keturgubo pagrindinai bei aštuongubo tikslumo ir ilgesni mainų standartai (IEEE 2008)
- Specialios reikšmės: neskaičiai (NaN), begalybės ($\pm\infty$), nuliai (± 0)
- Denormalizuoti skaičiai
- Apvalinimo valdymas
- Maskuojamos išimtinės situacijos
- Nustato operacijų tikslumą

IEEE 754 standarto kodavimas

- Mantisė: *absoliutus dydis su ženklu*
- Eksponentė: *skaičius su postūmiu* (postūmis $= 2^{n-1} - 1$ n bitų eksponentei)
- Eksponentės diapazonas: $-(2^{n-1} - 2) - +(2^{n-1} - 1)$ (pvz. 8 bitų eksponentei: $-126 - +127$)
- Trupmena (mantisė): normalizuotiems skaičiams „paslėptas“ bitas

$$0.1_{10} \approx 1.10011001100110011001101_2 \times 2^{-4}$$

Pavyzdys: 0.1 viengubo tikslumo s.k.:

p: $23 + 1$ bitas e: $-126 - 127$ (8 bitai); postūmis = $2^{8-1} - 1 = 128 - 1 = 127$ $f =$ `1.10011001100110011001101`
$e =$ `127 + (-4) = 123`$_{10}$ `= 01111011`$_2$

`0 01111011 10011001100110011001101`

Normalizuoti skaičiai

$$0.15625_{10} = \underbrace{0.00101_2}_{\text{nenormalizuotas}} = \pm 1.01 \times 2^{-3}$$

$$e = 127 + (-3) = 124_{10} = 01111100_2$$

Atvaizdavimas:
Float 32 (float; single precision):

$$\texttt{0 01111100 01 } \underbrace{\texttt{00...0}}_{\texttt{21 nulis}}$$

Denormalizuoti skaičiai

$$1.0_2 \times 2^{-130_{10}}$$

Viengubo tikslumo slankiam kableliui,

$$e_{\min} = -126_{10} \Rightarrow \text{ Neįmanoma normalizuoti!}$$

Atkreipkite dėmesį, kad:

$$e_{\min} + \text{ postūmis } = 127_{10} + (-126_{10}) = 1 = \texttt{0000\_0001}_2$$

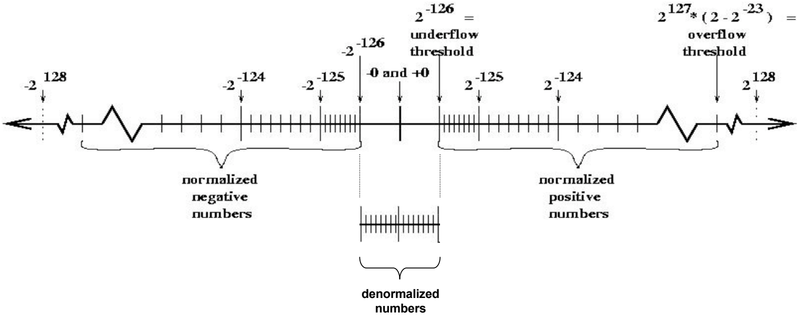
Pastumtai eksponentei `0000_0000`, interpretacija pasikeičia:

$$\texttt{0 0000 0000 0001}\underbrace{\texttt{00}\ldots\texttt{0}}_{\text{19 nulių}} = +0.0001_2 \times 2^{\mathbf{-126_{10}}} = +0.0625_{10} \times 2^{\mathbf{-126_{10}}}$$

eksponentė yra -126, **ne** -127 !

Kam reikalingi denormalizuoti skaičiai

Palaipsninis tikslumo praradimas



(Engelen 2008)

```
if (a != b) { x = a/(a-b); }
```

Nuliai

$$\mathtt{0\ 0000\,0000}\ \underbrace{\mathtt{00\ldots0}}_{\text{23 nuliai}} = \mathtt{0}$$

$$\mathtt{1\ 0000\,0000}\ \underbrace{\mathtt{00\ldots0}}_{\text{23 nuliai}} = -\mathtt{0}$$

$$\frac{1}{0} = \infty$$

$$\frac{1}{-0} = -\infty$$

```
if (a > b) { x = log(a-b); }
```

(Engelen 2008)

Begalybės

Liko nepanaudota eksponentė su visais vienetais, `1111_1111`

$$\texttt{0}\ \texttt{1111}\,\texttt{1111}\ \underbrace{\texttt{00}\ldots\texttt{0}}_{\text{23 nuliai}} = \infty$$

$$\texttt{1}\ \texttt{1111}\,\texttt{1111}\ \underbrace{\texttt{00}\ldots\texttt{0}}_{\text{23 nuliai}} = -\infty$$

$$\frac{1}{0} = +\infty; \qquad \frac{1}{+\infty} = +0$$

$$\frac{1}{-0} = -\infty; \qquad \frac{1}{-\infty} = -0$$

Neskaičiai: NaN

$$\texttt{0 1111 1111} \quad \underbrace{\mathbf{1}\texttt{1...0}}_{\text{ne visi 23 bitai yra nuliai}} = \text{qNaN}$$

$$\texttt{0 1111 1111} \quad \underbrace{\mathbf{0}\texttt{1...0}}_{\text{ne visi 23 bitai yra nuliai}} = \text{sNaN}$$

Operacijos, kurios grąžina NaN:

Operacija	NaN grąžina
$+$	$\infty + (-\infty)$
$\times$	$0 \times \infty$
$/$	$0/0,\ \infty/\infty$
rem	$0 \text{ rem } 0,\ \infty \text{ rem } \infty$
$\sqrt{}$	$\sqrt{x} \quad \forall x < 0$

(Goldberg 1991)

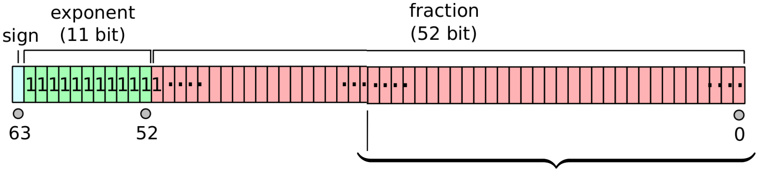
NaN palyginimas

- Bet koks palyginimas su NaN grąžina `False`, todėl kai $x <$ NaN yra neteisingas, dar nereiškia kad $x >=$ NaN
- `!(x < y)` $\not\Leftrightarrow$ `x >= y`
- `(x == y) == FALSE` kai `x == NaN`
- Negalima surikiuoti realių skaičių masyvo su NaN

(Engelen 2008)

NaN panaudojimas

Viengubo tikslumo skaičiams, NaN reikšmės turi 21 “laisvų” bitų
Dvigubo tikslumo skaičiams, NaN reikšmės turi 50 “laisvų” bitų



mažiausiai 32 bitai laisvi! Užtenka 32 bitų adresui...

- Dinaminės kalbos (pvz. JavaScript) naudoja “boxed NaN” reikšmes
- sNaN naudingi neinicializuotoms reikšmėms pagauti
- qNaN gali atvaizduoti nežinomas reikšmes

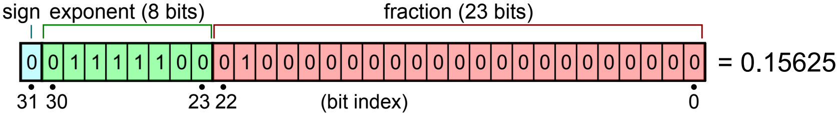
Apibendrinimas: IEEE 754 specialios reikšmės

Eskponentė	Trupmena	Reiškia
$e = e_{\text{min}} - 1$	$f = 0$	± 0
$e = e_{\text{min}} - 1$	$f \neq 0$	$\pm 0.f \times 2^{e_{\text{min}}}$
$e_{\text{min}} \leq e \leq e_{\text{max}}$	$f = \forall n$	$\pm 1.f \times 2^{e}$
$e = e_{\text{max}} + 1$	$f = 0$	$\pm\infty$
$e = e_{\text{max}} + 1$	$f \neq 0,$	NaN

(Goldberg 1991)

Viengubo tikslumo skaičiai

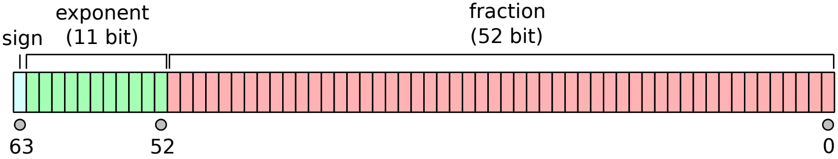
32-bitų skaičius



Vectorization: Stannered, CC BY-SA 3.0 via Wikimedia Commons

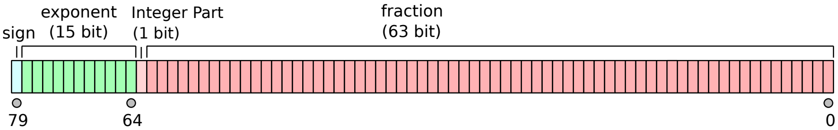
Dvigubo tikslumo skaičiai

64-bitų skaičius



https://en.wikipedia.org/wiki/File:IEEE_754_Double_Floating_Point_Format.svg

Intel 80 bitų išplėsto tikslumo skaičiai



BillF4, CC BY-SA 3.0, via Wikimedia Commons

- Nėra paslėpto bito
- Pakankamas tikslumas suskaičiuoti x^y
- Skirti tarpiniams rezultatams

Intel x87 slankaus kablelio registrai

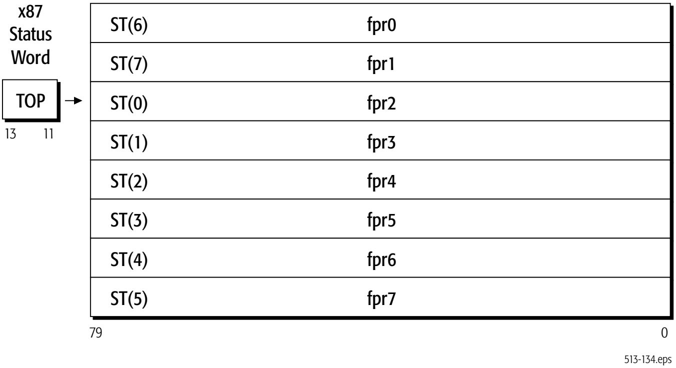


Figure 6-2. x87 Physical and Stack Registers

(AMD 2017)

Slankaus kablelio būsenos registrai

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B	C3	TOP			C2	C1	C0	ES	SF	PE	UE	OE	ZE	DE	IE

Bits	Mnemonic	Description
15	B	x87 Floating-Point Unit Busy
14	C3	Condition Code
13:11	TOP	Top of Stack Pointer 000 = FPR0 111 = FPR7
10	C2	Condition Code
9	C1	Condition Code
8	C0	Condition Code
7	ES	Exception Status
6	SF	Stack Fault
	Exception Flags	
5	PE	Precision Exception
4	UE	Underflow Exception
3	OE	Overflow Exception
2	ZE	Zero-Divide Exception
1	DE	Denormalized-Operand Exception
0	IE	Invalid-Operation Exception

Figure 6-3. x87 Status Word Register (FSW)

(AMD 2017)

Slankaus kablelio valdymo registrai

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			Y	RC		PC		Res		PM	UM	OM	ZM	DM	IM

Bits	Mnemonic	Description
12	Y	Infinity Bit (80287 compatibility)
11:10	RC	Rounding Control
9:8	PC	Precision Control
		#MF Exception Masks
5	PM	Precision Exception Mask
4	UM	Underflow Exception Mask
3	OM	Overflow Exception Mask
2	ZM	Zero-Divide Exception Mask
1	DM	Denormalized-Operand Exception Mask
0	IM	Invalid-Operation Exception Mask

Figure 6-4. x87 Control Word Register (FCW)

(AMD 2017)

Slankaus kablelio savybės

- Garantuotas atskirų operacijų tikslumas
  Except where stated otherwise, every operation shall be performed as if it first produced an intermediate result correct to infinite precision and with unbounded range, and then rounded that result according to one of the attributes in this clause.

  (IEEE 2019), sect. 4.3

- Kiekvienas išreiškiamas skaičius atvaizduojamas vieninteliu būdu
- FP skaičiai surikiuoti kaip sveiki skaičiai modulio su ženklu atvaizdavime!
  All of the possible single-precision entities are well ordered in the natural lexicographic ordering of their machine representations interpreted as sign-magnitude binary integers

  (Cody 1981)

Pavyzdys: 16-bitų SK kodas

```
gcc -c -S \
   -m16 -O3 --omit-frame-pointer \
   -o single-precision.asm single-precision.c
```

```c
float parallel( float x, float y )
{
    return x*y/(x + y);
}
```

```
parallel:
.LFB0:
        .cfi_startproc
        flds    4(%esp)
        flds    8(%esp)
        fld     %st(1)
        fmul    %st(1), %st
        fxch    %st(2)
        faddp   %st, %st(1)
        fdivrp  %st, %st(1)
        ret
```

Pavyzdys: 64-bitų SK kodas

```
gcc -c -S \
   -O3 --omit-frame-pointer \
   -o single-precision.asm single-precision.c
```

```c
float parallel( float x, float y )
{
    return x*y/(x + y);
}
```

```
parallel:
.LFB0:
        .cfi_startproc
        movaps  %xmm0, %xmm2
        addss   %xmm1, %xmm0
        mulss   %xmm1, %xmm2
        divss   %xmm0, %xmm2
        movaps  %xmm2, %xmm0
        ret
```

Alternatyvos

- Racionalių skaičių aritmetika
- Kintamo ilgio slankaus kablelio aritmetika
- J. Gustafsono Unum skaičių sistema (Gustafson 2015)
- Logaritminės skaičių sistemos (Coleman et al. 2008; Ismail et al. 2011)

Apibendrinimas

- Slankaus kablelio skaičiai yra realių skaičių artiniai
- Įprastose situacijose naudojami normalizuoti skaičiai
- Naudojami specialūs kodai denormalizuotiems skaičiams, begalybei, NaN ± 0
- Kiekvienas IEEE 754 slankaus kablelio (s.k.) objektas turi unikalų atvaizdavimą, ir kiekvienas dvejetainis kodas vaizduoja s.k. objektą
- Kai kurie s.k. objektai (pvz. NaN) turi savybes, kurios skiriasi nuo įprastų realių skaičių savybių
- Aktyviai tyrinėjamos naujos s.k. skaičių alternatyvos

Šaltiniai

AMD (Dec. 2017). *AMD64 Architecture Programmer's Manual, Volume 1: Application Programming, revision 3.22*. AMD. URL: `https://www.amd.com/system/files/TechDocs/24592.pdf`.

Cody, W. J. (Mar. 1981). "Analysis of proposals for the floating-point standard". In: *Computer* 14.3, pp. 63–68. DOI: `10.1109/c-m.1981.220379`.

Coleman, John N. et al. (2008). "The European Logarithmic Microprocesor". In: *IEEE Transactions on Computers* 57.4, pp. 532–546. DOI: `10.1109/tc.2007.70791`.

Engelen, Robert van (2008). *Floating point operations and SIMD extensions*. URL: `http://www.cs.fsu.edu/~engelen/courses/HPC-adv-2008/FP.pdf`.

Goldberg, David (1991). "What every computer scientist should know about floating-point arithmetic". In: *ACM Comput. Surv.* 23, pp. 5–48. ISSN: 0360-0300. DOI: `10.1145/103162.103163`. URL: `http://doi.acm.org/10.1145/103162.103163`.

Gustafson, John L. (Aug. 2015). *The End of Error Unum Computing by Gustafson, John L.* Vol. 1. CRC Press. ISBN: 978-14-8223-987-4.

IEEE (Oct. 1985). *IEEE standard for binary floating-point arithmetic*. IEEE. DOI: `10.1109/ieeestd.1985.82928`.

— (2008). *IEEE standard for floating-point arithmetic*. DOI: `10.1109/ieeestd.2008.4610935`.

— (2019). *IEEE standard for floating-point arithmetic*. IEEE. DOI: `10.1109/ieeestd.2019.8766229`.

Ismail, R. Che et al. (July 2011). "ROM-less LNS". In: *2011 IEEE 20th Symposium on Computer Arithmetic*. IEEE, pp. 43–51. DOI: `10.1109/arith.2011.15`.