

Systemy liczenia

- dwójkowy (binarny), ósemkowy, dziesiętny, szesnastkowy.

system	cyfry	przykład	dziesiętnie
dwójkowy	0 1	101	$1 * 2^2 + 1 * 2^0 =$ $1 * 4 + 1 = 5$
ósemkowy	0 1 2 3 4 5 6 7	270	$2 * 8^2 + 7 * 8^1 =$ $2 * 64 + 7 * 8 = 184$
dziesiętny	0 1 2 3 4 5 6 7 8 9	309	$3 * 10^2 + 9 * 10^0 =$ $3 * 100 + 9 = 309$
szesnastkowy	0 1 2 3 4 5 6 7 8 9 A B C D E F	10A	$1 * 16^2 + 10 * 16^0 =$ $1 * 256 + 10 = 266$

Komputer „myśli” dwójkowo – organizacja pamięci

- 1 **bit** – miejsce na zapamiętanie znaku 0 bądź 1
- 1 **bajt** = 8 bitów – pozwala zapamiętać 2^8 różnych wartości, a więc np. liczby 0-255
- 1 **kB** (kilobajt) = 2^{10} B = 1024 B
- 1 **MB** (megabajt) = 2^{20} B = 1048576 B
- 1 **GB** (gigabajt) = 2^{30} B = 1073741824 B
- 1 **słowo** to ilość informacji przetwarzanej przez procesor w jednym cyklu (procesor 32 bitowy = procesor o słowie 32 bitowym)
- bity i bajty bardziej i mniej znaczące

Binarna reprezentacja danych

- operacje bitowe

- przesunięcia:

$$00000001 \ll 3 = 00001000 = 8$$

$$00001101 \gg 2 = 00000011 = 3$$

$$00001101 \ll 5 = 10100000 = 160$$

- operacje logiczne: koniunkcja &, alternatywa |, dysjunkcja ^, dopełnienie ~

$$01010101 \& 11001100 = 01000100$$

$$01010101 | 11001100 = 11011101$$

$$00001101 \wedge 11001100 = 11000001$$

$$\sim 01010101 = 10101010$$

$$\sim 11001100 = 00110011$$

- typowa liczba całkowita zajmuje jedno słowo

Reprezentacja liczb

- Reprezentacja stałopozycyjna
 - w praktyce tylko dla liczb całkowitych
- Reprezentacja zmiennopozycyjna (zmiennoprzecinkowa)
 - mantysa i cecha, liczba = mantysa * 2^{cecha}
 $18.5 = 0.185 * 10^2 = 0.100101 * 2^{101}$
 - rozdzielczość i skończoność komputerowych liczb rzeczywistych
 - typy 4-bajtowe (3 bajty mantysy i 1 bajt cechy), 6-bajtowe, 8-bajtowe (podwójnej precyzji)
 - należy być ostrożnym przy dodawaniu małej liczby zmiennoprzecinkowej do dużej (mała może zniknąć)
 - wartości, które nie są poprawnymi liczbami (NaN - not a number)
 - standard IEEE 754 – dobry opis pod <http://research.microsoft.com/~hollasch/cgindex/coding/ieeefloat.html>

Standard IEEE 754

Reprezentacja bitowa

precyzja	znak	wykładnik	część ułamkowa	przesunięcie
pojedyncza	1 [31]	8 [30-23]	23 [22-00]	127
podwójna	1 [63]	11 [62-52]	52 [51-00]	1023

- reprezentacja binarna – podstawa równa 2
- znak – 0 - liczba dodatnia, 1 - liczba ujemna
- wykładnik i przesunięcie – rzeczywista cecha = wykładnik - przesunięcie
- mantysa
 - *postać znormalizowana liczby* – w mantysie przecinek po pierwszej niezerowej cyfrze ($1 \leq m < 10$)
Przykład: $1.25 * 10^2$, nie $0.125 * 10^3$ ani $125 * 10^0$
 - w systemie dwójkowym niezerowa znaczy jedynka
 - pierwszy bit mantysy zawsze równy 1 – ukryty, reszta to część ułamkowa

Standard IEEE 754 – przykłady

23.75

- postać binarna: $10111.11 = 2^4 + 2^2 + 2^1 + 2^0 + 2^{-1} + 2^{-2}$
- po normalizacji: $1.011111 * 2^4 = 1.011111 * 2^{100}$
- znak 0
wykładnik z przesunięciem $4 + 127 = 131 = 10000011$
część ułamkowa 011111
- ostatecznie mamy 0 10000011 011111000000000000000000
czyli 01000001 10111110 00000000 00000000
szesnastkowo 0x41BE0000

-0.7

- postać binarna: $-0.1011001100110011\dots$
- znak = 1, wykładnik = -1 , $127 - 1 = 126 = 01111110$
- ostatecznie mamy 1 01111110 01100110011001100110011
czyli 10111111 00110011 00110011 00110011
szesnastkowo 0xBF333333

Standard IEEE 754 – c.d.

Wartości specjalne – wykładniki z samych jedynek bądź z samych zer

znak	wykładnik (w)	cz. ułamkowa (u)	wartość
0	00..00	00..00	+0
0	00..00	00..01 – 11..11	dodatnia zdenorm. $0.u * 2^{-p+1}$
0	00..01 – 11..10	00..00 – 11..11	dodatnia znorm. $1.u * 2^{w-p}$
0	11..11	00..00	+Infinity (nieskończoność)
0	11..11	00..01 – 01..11	SNaN (Signalling Not A Number)
0	11..11	10..00 – 11..11	QNaN (Quiet Not A Number)
1	00..00	00..00	-0
1	00..00	00..01 – 11..11	ujemna zdenorm. $-0.u * 2^{-p+1}$
1	00..01 – 11..10	00..00 – 11..11	ujemna znorm. $-1.u * 2^{w-p}$
1	11..11	00..00	-Infinity (nieskończoność)
1	11..11	00..01 – 01..11	SNaN
1	11..11	10..00 – 11..11	QNaN

Standard IEEE 754 – c.d.

Operacje specjalne:

Operacja	Wynik
$n / \pm\text{Infinity}$	0
$\pm\text{Infinity} \times \pm\text{Infinity}$	$\pm\text{Infinity}$
$\pm\text{nonzero} / 0$	$\pm\text{Infinity}$
$\text{Infinity} + \text{Infinity}$	Infinity

Operacja	Wynik
$\pm 0 / \pm 0$	NaN
$\text{Infinity} - \text{Infinity}$	NaN
$\pm\text{Infinity} / \pm\text{Infinity}$	NaN
$\pm\text{Infinity} \times 0$	NaN

Zakresy dwójkowo:

precyzja	zdenormalizowane	znormalizowane
pojedyncza	$\pm 2^{-149} \dots (1 - 2^{-23}) * 2^{-126}$	$\pm 2^{-126} \dots (2 - 2^{-23}) * 2^{127}$
podwójna	$\pm 2^{-1074} \dots (1 - 2^{-52}) * 2^{-1022}$	$\pm 2^{-1022} \dots (2 - 2^{-52}) * 2^{1023}$

Zakresy dziesiętne:

precyzja	dziesiętne
pojedyncza	$\pm \sim 10^{-44.85} \dots \sim 10^{38.53}$
podwójna	$\pm \sim 10^{-323.3} \dots \sim 10^{308.3}$