SYSTEMY LICZBOWE

SYSTEMY POZYCYJNE:

- dziesiętny (arabski): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- rzymski: I, II , III, V, C, M

System pozycyjno–wagowy: na przykład liczba 444

4 4 4

$4*10$ $4*10$ $4*1$

Wagi systemu dziesiętnego: 1, 10, 100, 1000, ...

$$\boxed{L = C_{n-1} \cdot P^{n-1} + C_{n-2} \cdot P^{n-2} + \cdots + C_1 \cdot P^1 + C_0 \cdot P^0}$$

C – elementy zbioru cyfr dostępnych w danym systemie,

$$C \in \{0,...,P-1\},$$

P – podstawa systemu, P = 2, 4, 8, 10, 16 (60 – Babilon, czas),
n – liczba całkowita.

<u>Przykłady</u>:

$P = 2 \rightarrow \quad C \in \{0,1\}$

$P = 4 \rightarrow \quad C \in \{0,1,2,3\}$

$P = 8 \rightarrow \quad C \in \{0,1,2,3,4,5,6,7\}$

$P = 10 \rightarrow \quad C \in \{0,1,2,3,4,5,6,7,8,9\}$

$P = 16 \rightarrow \quad C \in \{\underbrace{0,1,2,3,4,5,6,7,8,9}_{10\ \text{cyfr}},\underbrace{A,B,C,D,E,F}_{\text{uzupełnienie}}\}$

ZAPIS liczby **1011** w różnych systemach (n = 4):

$1011_{(2)} = 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8+0+2+1 = 11$

$1011_{(4)} = 1 \cdot 4^3 + 0 \cdot 4^2 + 1 \cdot 4^1 + 1 \cdot 4^0 = 64+0+4+1 = 69$

$1011_{(8)} = 1 \cdot 8^3 + 0 \cdot 8^2 + 1 \cdot 8^1 + 1 \cdot 8^0 = 512+0+8+1 = 521$

$1011_{(10)} = 1 \cdot 10^3 + 0 \cdot 10^2 + 1 \cdot 10^1 + 1 \cdot 10^0 = 1000+0+10+1 = 1011$

$1011_{(16)} = 1 \cdot 16^3 + 0 \cdot 16^2 + 1 \cdot 16^1 + 1 \cdot 16^0 = 4096+0+16+1 = 4113$

System (10)	→	SYSTEM NATURALNY
System (2)	→	SYSTEM KOMPUTEROWY {0, 1}
System (16)	→	SYSTEM KOMPUTEROWY

Przykłady:

$10_{(16)} = 16_{(10)}$

$$\boxed{\begin{aligned} 10_{(16)} &= 1 \cdot 16^1 + 0 \cdot 16^0 = 16 + 0 = 16 \\ 16_{(10)} &= 1 \cdot 10^1 + 6 \cdot 10^0 = 10 + 6 = 16 \end{aligned}}$$

$19_{(16)} = 25_{(10)}$

$$\boxed{\begin{aligned} 19_{(16)} &= 1 \cdot 16^1 + 9 \cdot 16^0 = 16 + 9 = 25 \\ 25_{(10)} &= 2 \cdot 10^1 + 5 \cdot 10^0 = 20 + 5 = 25 \end{aligned}}$$

$BF_{(16)} = 191_{(10)}$

$$\boxed{\begin{aligned} BF_{(16)} &= B \cdot 16^1 + F \cdot 16^0 = 11 \cdot 16 + 15 \cdot 1 = 191 \\ 191_{(10)} &= 1 \cdot 10^2 + 9 \cdot 10^1 + 1 \cdot 10^0 = 100 + 90 + 1 = 191 \end{aligned}}$$

SYSTEM DZIESIĘTNY

Podstawa P = 10, znaki **+** oraz **–**

Liczby są przedstawiane jako sumy potęg podstawy 10.

$$\boxed{1245{,}245 = 1 \cdot 10^3 + 2 \cdot 10^2 + 4 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1} + 4 \cdot 10^{-2} + 5 \cdot 10^{-3}}$$

SYSTEM DWÓJKOWY (BINARNY)

Znaki: 0, 1

Dwójkowy system pozycyjny, kod dwójkowy

ZALETY:
- prostota
- łatwa realizacja techniczna (elektronika)
- możliwość interpretacji cyfr {0, 1} jako wartości logicznych (**algebra Boole'a**)

WADY:
- długość zapisu
- przyzwyczajenie

KOMPUTERY: konwersja systemów 10 → 16 → 2

Liczby w zapisie dwójkowym:

$$c_{n-1}\ c_{n-2}\ \ldots\ldots\ c_1\ c_0$$

c_n, c_{n-1} – cyfry liczby dwójkowej, przyjmujące wartości 0, 1

Wartość liczby binarnej w systemie dziesiętym:

$$L = c_{n-1} \cdot 2^{n-1} + c_{n-2} \cdot 2^{n-2} + \cdots + c_1 \cdot 2^1 + c_0 \cdot 2^0$$

Porównanie systemów P = 10 oraz P = 2:

	System dziesiętny P = 10	System dwójkowy P = 2					
n = 16 pozycji	0						0
	1						1
	2					1	0
	3					1	1
	4				1	0	0
	5				1	0	1
	6				1	1	0
	7				1	1	1
	8			1	0	0	0
	9			1	0	0	1
	10			1	0	1	0
	11			1	0	1	1
	12			1	1	0	0
	13			1	1	0	1
	14			1	1	1	0
	15			1	1	1	1
	16		1	0	0	0	0
	....	....					

LICZBY NATURALNE (CAŁKOWITE DODATNIE)

$$\underset{4\,3\,2\,1\,0}{\underline{10100}} \rightarrow \underline{1 \cdot 2^4} + 0 \cdot 2^3 + \underline{1 \cdot 2^2} + 0 \cdot 2^1 + 0 \cdot 2^0 = 16 + 4 = 20$$

$n = 5$

$$100101 \rightarrow \underline{1 \cdot 2^5} + 0 \cdot 2^4 + 0 \cdot 2^3 + \underline{1 \cdot 2^2} + 0 \cdot 2^1 + \underline{1 \cdot 2^0} =$$
$$= 32 + 4 + 1 = 37$$

KONWERSJA LICZBY DZIESIĘTNEJ DO DWÓJKOWEJ:

$$(147)_{10} = (\ ?\ )_2$$

	Reszta:	
147 : 2 = 73	$C_0 = 1$	
73 : 2 = 36	$C_1 = 1$	
36 : 2 = 18	$C_2 = 0$	
18 : 2 = 9	$C_3 = 0$	n = 8
9 : 2 = 4	$C_4 = 1$	
4 : 2 = 2	$C_5 = 0$	
2 : 2 = 1	$C_6 = 0$	
1 : 2 = 0	$C_7 = 1$	

$$(147)_{10} = (10010011)_2$$

$$10010011 = 1 \cdot 2^7 + 1 \cdot 2^4 + 1 \cdot 2^1 + 1 \cdot 2^0 =$$
$$= 128 + 16 + 2 + 1 = 147$$

$127_{(10)} = 1111111_{(2)}$ $\quad n = 7$

$243_{(10)} = 11110011_{(2)}$ $\quad n = 8$

$$\boxed{\begin{array}{c} 11111111_{(2)} = 2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 = \\ = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255_{(10)} \end{array}}$$

Zapis binarny prosty pozwala za pomocą n cyfr zapisywać liczby z zakresu:

$$\boxed{\mathbf{0 \leq L_{10} \leq 2^n - 1}}$$

$$\boxed{\text{Dla } \mathbf{n = 8}\text{:} \quad 0 \leq L_{10} \leq 2^8 - 1 = 256 - 1 = 255}$$

KODOWANIE DODATNICH LICZB UŁAMKOWYCH

(ułamki właściwe)

$$\boxed{c_{-1}\ c_{-2}\ \ldots.\ c_{-n}}$$

$$L_{10} = c_{-1} \cdot 2^{-1} + c_{-2} \cdot 2^{-2} + \ldots + c_{-n} \cdot 2^{-n}$$

$$1011_{(2)} = 1\cdot 2^{-1} + 0\cdot 2^{-2} + 1\cdot 2^{-3} + 1\cdot 2^{-4} =$$

$$= \frac{1}{2} + \frac{1}{8} + \frac{1}{16} = \frac{8+2+1}{16} = \frac{11}{16} = 0{,}6875_{(10)}$$

n = 8:

$$11111111_{(2)} = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{64} + \frac{1}{128} + \frac{1}{256} = \frac{255}{256} = 0{,}99609375$$

KODOWANIE DOWOLNYCH LICZB I UŁAMKÓW

– należy wprowadzić znak „–” dla liczb ujemnych

– nalezy wprowadzić znak oddzielający część całkowitą liczby od części ułamkowej

0 – ułamek nieujemny (dodatni) 1 – ułamek niedodatni (ujemny)

KODOWANIE BEZPOŚREDNIE

KODOWANIE ODWROTNE

KOD UZUPEŁNIENIOWY

Liczby rzeczywiste

Liczby rzeczywiste – część całkowita + część ułamkowa

Zapis w dwóch bajtach (16 liczb):

PRZECINEK

1	1	1	1	1	1	1	1	,	1	1	1	1	1	1	1	

n = 8 n = 8

$$2^7+2^6+2^5+2^4+2^3+2^2+2^1+2^0 = 255$$

$$2^{-1}+2^{-2}+2^{-3}+2^{-4}+2^{-5}+2^{-6}+2^{-7}+2^{-8} = \frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\frac{1}{32}+\frac{1}{64}+\frac{1}{128}+\frac{1}{256}=\frac{255}{256}$$

Powyższy zapis ma same wady:

1. nie można zapisać liczb większych od 255,
2. przy małych liczbach pozostaje dużo wolnego miejsca,
3. MARNOWANIE PAMIĘCI KOMPUTEROWEJ.

Błąd przy zapisie liczb:

	Liczba dziesiętna	Część całkowita									Część ułamkowa									
1.	128	1	0	0	0	0	0	0	0	,	0	0	0	0	0	0	0	0	0	1
2.	1	0	0	0	0	0	0	0	1	,	0	0	0	0	0	0	0	0	0	1
3.	1/256	0	0	0	0	0	0	0	0	,	0	0	0	0	0	0	0	0	1	1

„Obcięcie” liczb na dziewiątym miejscu – błąd bezwzględny

$2^{-9} = \frac{1}{2^9} = \frac{1}{512} = 0{,}001953125$ - wartość tracona z powodu braku miejsca.

BŁĄD WZGLĘDNY:

Liczba 1: $\cong$ 0,0015%

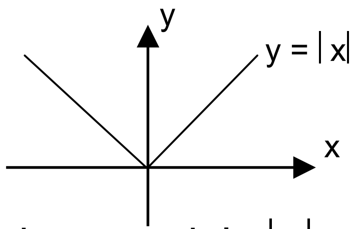
Liczba 2: $\cong$ 0,1945%

Liczba 3: $\cong$50% **!**

Powyższy sposób zapisu powoduje,
że obliczenia są niewiarygodne
(obliczenia naukowe, ekonomiczne, multimedialne).

**ZAPIS STAŁOPRZECINKOWY
DLA LICZB RZECZYWISTYCH
JEST NIEPRAKTYCZNY.**

Kodowanie liczb o module $|x| > 1$



$$y = \begin{cases} -x & \text{dla } x < 0 \\ x & \text{dla } x \geq 0 \end{cases}$$

Liczby o module $|x| > 1$ przedstawia się w postaci składającej się z dwóch części:

– ułamkowej $\rightarrow$ **mantysa** (m)

– całkowitej $\rightarrow$ **cecha** (c)

Wartość liczby określa zależność:

$$x = \text{sign}(x) \cdot m \cdot 2^c \qquad 0 \leq m \leq 1$$

$$\text{sign}(x) = \begin{cases} 1 & \text{dla } x \geq 0 \\ 0 & \text{dla } x < 0 \end{cases}$$

ZAPIS LICZBY x:

– liczby stałoprzecinkowe (fixed point)

– liczby zmiennoprzecinkowe (floating point)
(zmiennopozycyjne)

DEFINICJA:

BIT **– NAJMNIEJSZA JEDNOSTKA INFORMACJI {0, 1}**

kb	Mb	Gb	Tb
kilobit	megabit	gigabit	terabit

1 **bajt** = 8 bitów (*ang. byte*)

kB	MB	GB	TB
kilobajt	megabajt	gigabajt	terabajt

LICZBY STAŁOPRZECINKOWE

– oddzielne kodowanie modułu i cechy
– ustalenie stałej umownej pozycji przecinka, oddzielającego część całkowitą od ułamkowej

$2340{,}23 = 0{,}234023 \cdot 10^{4}$	cecha c = 4
$2{,}7363 = 0{,}27363 \cdot 10^{1}$	c = 1
$0{,}15934 = 0{,}15934 \cdot 10^{0}$	c = 0
$0{,}000243 = 0{,}243 \cdot 10^{-3}$	c = -3

Mantysa należy do przedziału (0, 1),
cecha pozwala przesunąć przecinek.

ZAKRES LICZB: max c = 128 (n = 8, c = 2^{7})
$L_{max} = 3{,}4 \cdot 10^{38}$

Można zwiększyć zapis cechy do 2 bajtów.

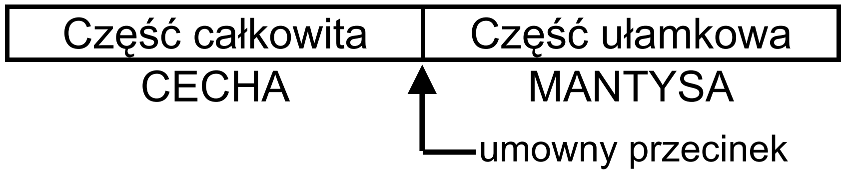
DEFINICJA:

SŁOWO KOMPUTEROWE

Ilość informacji przetwarzanej przez komputer.

KOMPUTER 8–, 16–, 32– (64–, 128-) bitowy oznacza wielkość grupy danych, którą komputer może operować jako całością.

Słowo:

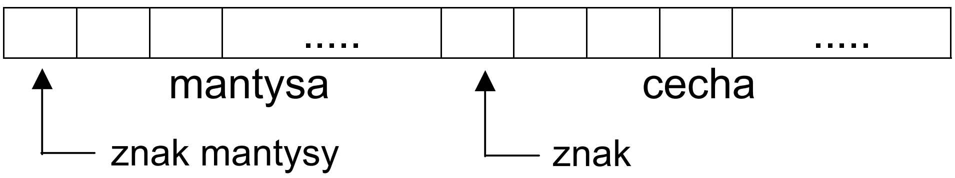


Zalety zapisu stałoprzecinkowego:
– prostota
– elastyczność

Wady zapisu stałoprzecinkowego:
– ograniczony zakres liczb
– mała dokładność obliczeń (zaokrąglanie wyników)

LICZBY ZMIENNOPRZECINKOWE

Liczby kodowane są w jednym słowie.



Mantysa – trzy bajty, cecha – jeden bajt

Wartość liczby $\boxed{L = m \cdot 2^c}$

Zalety zapisu zmiennoprzecinkowego:
– szeroki zakres liczb

Wady zapisu stałoprzecinkowego:
– zajmowanie dużej pamięci

STANDARD ZAPISU LICZB ZMIENNOPRZECINKOWYCH

KOD UZUPEŁNIEŃ DO 2

Zapis tradycyjny: 8 bitów, c_0, ..., c_7 przyjmują wartości 0, 1

$$\boxed{L = c_7 \cdot 2^7 + c_6 \cdot 2^6 + c_5 \cdot 2^5 + c_4 \cdot 2^4 + c_3 \cdot 2^3 + c_2 \cdot 2^2 + c_1 \cdot 2^1 + c_0 \cdot 2^0}$$

Dla zapisu liczb ujemnych przyjmuje się: $\mathbf{c_7 \cdot (-2^7) = c_7 \cdot (-128)}$

$$\boxed{L = c_7 \cdot (-2^7) + c_6 \cdot 2^6 + c_5 \cdot 2^5 + c_4 \cdot 2^4 + c_3 \cdot 2^3 + c_2 \cdot 2^2 + c_1 \cdot 2^1 + c_0 \cdot 2^0}$$

$C_7 = 1 \rightarrow$ liczba ujemna

11111111 = -128 +64+32+16+8+4+2+1 = -128 + 127 =-1

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Bit najstarszy **Bit najmłodszy**

Bit najstarszy MSB (*Most Significant Bit*)
Bit najmłodszy LSB (*Least Significant Bit*)

BIT7 służy do kodowanie znaku
BIT7 = 0 $\rightarrow$ liczba dodatnia
BIT7 = 1 $\rightarrow$ liczba ujemna

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	$\mathbf{L_{10}}$
0	0	0	0	1	0	0	0	**8**
0	0	0	0	1	1	0	0	**12**
1	0	0	0	1	1	0	0	**-116**
0	1	1	1	1	1	1	1	**127**
1	1	1	1	1	1	1	1	**-1**

SYSTEM SZESNASTKOWY (HEKSADECYMALNY)

	System dziesiętny P = 10	**System szes-nastkowy P = 16**	System dwójkowy P = 2
n = 16 pozycji	0	**0**	0
	1	**1**	1
	2	**2**	10
	3	**3**	11
	4	**4**	100
	5	**5**	101
	6	**6**	110
	7	**7**	111
	8	**8**	1000
	9	**9**	1001
	10	**A**	1010
	11	**B**	1011
	12	**C**	1100
	13	**D**	1101
	14	**E**	1110
	15	**F**	1111

Zaleta: prostsze przedstawianie liczb dwójkowych.

ZAMIANA LICZB BINARNYCH NA HEKSADECYMALNE:

podział liczby dwójkowej na grupy cyfr 4-bitowych

110	1011	1101	$= 6BD_{(16)}$
0110	1011	1101	
6	B	D	

Zapis „16” umożliwia przedstawianie wszystkich bitów w bajcie za pomocą dwóch znaków.

Konwersja binarno – szesnastkowa:

$1011 \mid 0011 \mid 1001 = B29_{(16)} = 2857_{(10)}$

$110 \mid 1011 \mid 1101_{(2)} = 6BD_{(16)} = 1725_{(10)}$

Konwersja szesnastkowo – binarna:

$3FA_{(16)} = 0011 \mid 1111 \mid 1010_{(2)} = 1018_{(10)}$

Zapis dwójkowy jest długi.
Zapis szesnastkowy stanowi kompromis pomiędzy tym co jest najbliższe komputerowi a tym co jest wygodne dla człowieka.