Ćwiczenie 1
Prezentacja danych w systemach komputerowych

Część 1

Dlaczego system binarny?

Pojęcie bitu

Bit – jednostka informacji wystarczająca do zakomunikowania jednego z dwu równo prawdopodobnych zdarzeń.

1

0

Przyczyny zastosowania systemu binarnego

kb	Mb	Gb	Tb
kilobit	megabit	gigabit	terabit

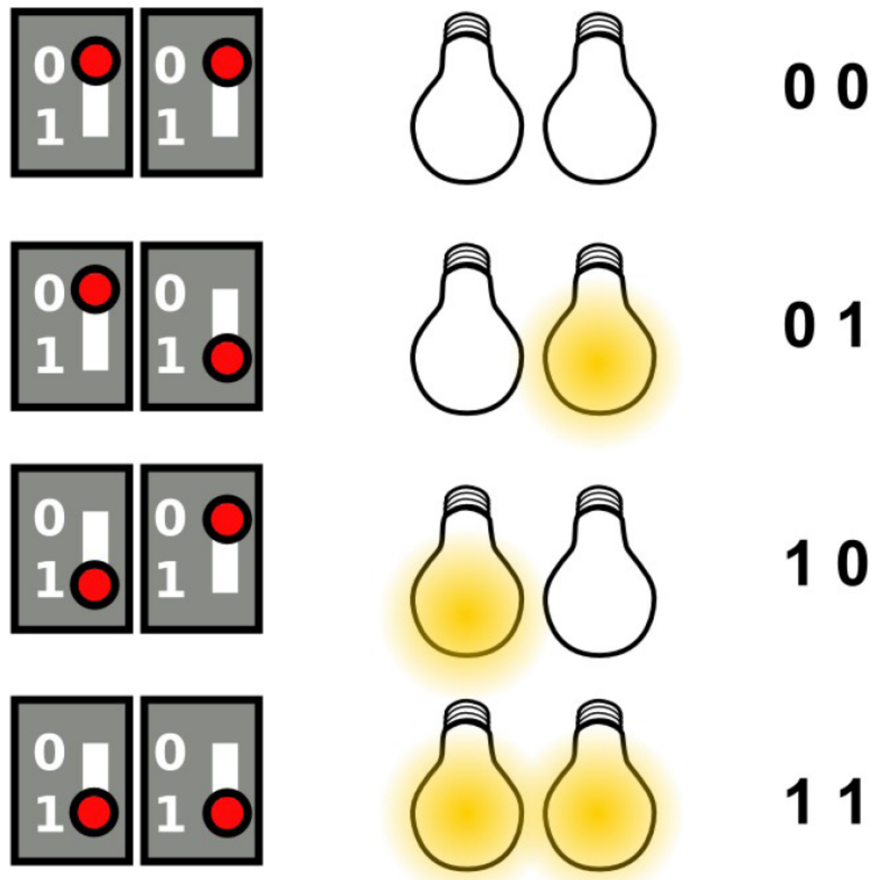
1 **bajt** = 8 bitów (*ang. byte*)

kB	MB	GB	TB
kilobajt	megabajt	gigabajt	terabajt

Przyczyny zastosowania systemu binarnego

Przyczyny zastosowania systemu binarnego w technologii cyfrowej to:

- łatwość implementacji elektrycznej i elektronicznej,
- odporność na zakłócenia,
- możliwość interpretacji cyfr {0, 1} jako wartości logicznych (algebra Boole'a).



Przyczyny zastosowania systemu binarnego

Ciekawostka:

Jedynym znanym komputerem zbudowanym z elementów

3-stanowych był eksperymentalny radziecki Sietuń (1959).

Element reprezentujący jednostkę informacji stanowiła para rdzeni magnetycznych, z których każdy mógł być namagnesowany w jednym z dwóch kierunków; czwarty -niewykorzystany stan -służył do celów kontrolnych.

Część 2

Pozycyjne systemy liczbowe

System dziesiętny

0 0 0
0 0 1
0 0 2
...
...
...
9 9 7
9 9 8
9 9 9

Ile różnych liczb można zapisać w systemie dziesiętnym za pomocą 3 cyfr?

System dziesiętny

000
001
002
...
...
...
997
998
999

Tysiąc – od 0 do 999

W dowolnym systemie liczbowym można przedstawić

$$P^n$$

P - podstawa systemu
n - ilość cyfr

liczb/kombinacji.

System dziesiętny

Przykładowa liczba 907 w systemie dziesiętnym powstaje wg poniższego schematu:

$$\begin{array}{ccc} 10^2 & 10^1 & 10^0 \\ 9 & \color{red}{0} & 7 \end{array} = 9*100+\color{red}{0*10}+7*1=907_{10}$$

System o dowolnej podstawie

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,

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

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

n – liczba całkowita.

System o dowolnej podstawie

Przykłady:

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

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

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

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

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

System o dowolnej podstawie

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

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

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

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

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

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

Systemy niepozycyjne

Zupełnie inna sytuacja występuje w zapisie liczby w systemie rzymskim.

Kolejne liczby od 1; : : : ; 9 mają postać:

I; II; III; IV;V;V I;V II;V III; IX

Widać, że w takim zapisie pozycja cyfry (o ile w ogóle można mówić w tym wypadku o cyfrze), nie jest związana z wyznaczaniem jej wartości, lecz istotna jest postać całej liczby.

Taki system zapisu nazywamy ***addytywnym systemem liczbowym.***

System dwójkowy (binarny)

Korzystając z definicji pozycyjnego systemu liczbowego otrzymujemy, że podstawą systemu dwójkowego jest liczba 2, oraz cyframi tego systemu są elementy zbioru <0; 1>.

Zapiszmy przykładową liczbą w tym systemie

$$x = 1011110110_{(2)}$$

otrzymujemy:

$$x = 1*2^9 + 0*2^8 + 1*2^7 + 1*2^6 + 1*2^5 +$$

$$+ 1*2^4 + 0*2^3 + 1*2^2 + 1*2^1 + 0*2^0$$

Zastępując teraz potęgi liczby 2 odpowiednimi wartościami, otrzymujemy

$$x = 1*512 + 0*256 + 1*128 + 1*64 + 1*32 +$$

$$+ 1*16 + 0*8 + 1*4 + 1*2 + 0*1 = 758_{(10)}$$

System dwójkowy (binarny)

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

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

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

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

System dwójkowy (binarny)

ZALETY:

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

WADY:

- długość zapisu
- przyzwyczajenie

Część II

Kodowanie liczb w systemach binarnych

Ważniejsze potęgi dwójki

$2^0=1$

$2^1=2$

$2^2=4$

$2^3=8$

$2^4=16$

$2^5=32$

$2^6=64$

$2^7=128$

$2^8=256=1$ bajt

1bajt=8bitów

10101111

$2^{16}=65.536$

$2^{24}=16.777.216$

2^{10}bajtów=1kB **(1024)**

2^{20}bajtów=1MB **(1024*1024)**

2^{30}bajtów=1GB **(1024*1024*1024)**

Ważniejsze potęgi dwójki

Kolor 24 / 32 bitowy, dźwięk 16 bitowy

$$2^{24} = 2^{10} * 2^{10} * 2^{4} = 1024 * 1024 * 16 \approx 16\ 000\ 000$$

$$2^{32} \approx 1\ 000\ 000\ 000 * 4$$

System dwójkowy - konwersja

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	$C_3 = 0$
9 : 2 = 4	$C_4 = 1$
4 : 2 = 2	$C_5 = 0$
2 : 2 = 1	$C_6 = 0$
1 : 2 = 0	$C_7 = 1$

n = 8

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

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

System dwójkowy - arytmetyka

Dodawanie w systemie dwójkowym

```
   11
  00111000       01000001
+ 00010001     + 00010100
----------     ----------
  01001001       01010101
```

Dodawanie jest realizowane podobnie jak dla systemu dziesiętnego, należy jedynie pamiętać, że

$$1_{(2)} + 1_{(2)} = 10_{(2)}$$

System dwójkowy - arytmetyka

Odejmowanie w systemie dwójkowym

```
  00111001     00101101
- 00001101   - 00010001
----------   ----------
  00101100     00011100
```

W przypadku odejmowania 0 - 1 w systemie dwójkowym, musimy dokonać zapożyczenia 1 na następnej pozycji liczby.

System dwójkowy - arytmetyka

Mnożenie w systemie dwójkowym

```
       1111
   x    101
-------------
       1111
      0000
+    1111
-------------
    1001011
```

```
      10001
   x     11
-------------
      10001
+    10001
-------------
     110011
```

Mnożenie jest wykonywane analogicznie jak mnożenie w systemie dziesiętnym.

System dwójkowy - arytmetyka

Dzielenie w systemie dwójkowym

```
     110
   -----
   10010:11=00000110
-    11
 -------
      1
      11
 -    11
 -------
       0
```

```
     1011
   -------
   1111001:1011=1011
 - 1011
 ---------
    10000
 -    1011
 ---------
       1011
 -     1011
 ---------
          0
```

Dzielenie podobnie jak mnożenie wykonujemy tak samo jak w przypadku dzielenia w systemie dziesiętnym.

System szesnastkowy (hexadecymalny)

Duże liczby binarne są nieczytelne.

01010010100100100001111001010100101010110

Celem wprowadzenia systemy szesnastkowego jest skrócenie zapisu bez przeliczania na system dziesiętny.

Każde 4 bity da się przedstawić za pomocą 1 cyfry szesnastkowej – bez żadnego przeliczania.

hex	bin	dec
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

System szesnastkowy (hexadecymalny)

Przykład:

```
0101001010010010000111100101010010101010101100

0101 0010 1001 0010 0001 1110 0101 0100 1010
1010 1100

0101 0010 1001 0010 0001 1110 0101 0100 1010
   5    2    8    2    1    E    5    4    A
1010 1100
   A    C

52821E54AAC
```

hex	bin	dec
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

System szesnastkowy (hexadecymalny)

System szesnastkowy podlega tym samym zasadą co inne systemy wagowo – pozycyjne .

16^5	16^4	16^3	16^2	16^1	16^0
4	**F**	**D**	**3**	**0**	**D**

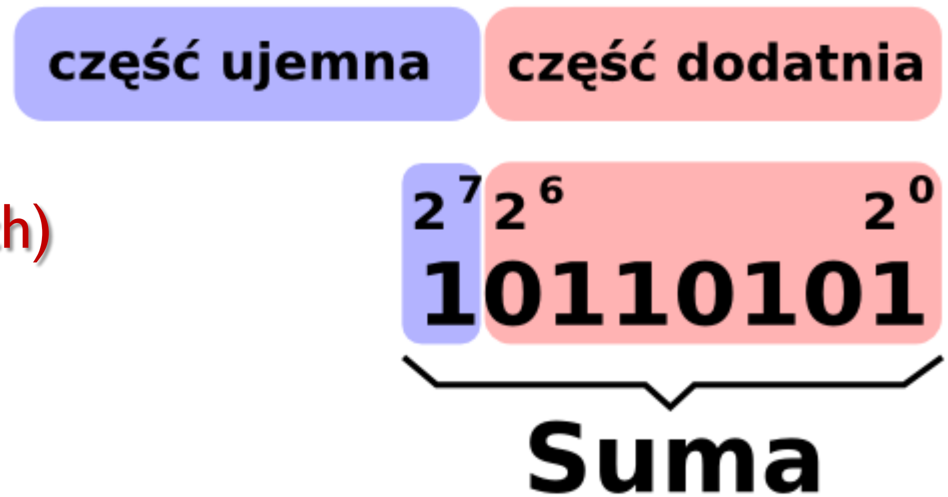
$$=4*16^5+15*16^4+13*16^3+$$

$$+3*16^2+0*16^1+13*16^0$$

hex	bin	dec
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Kodowanie liczb ujemnych

Kod U2
(Uzupełnień do dwóch)



Najmniejsza liczba

10000000=-128+0=-128

Największa liczba

01111111=-0+127=127

Kodowanie liczb ujemnych

Problem: wygenerować w KU2 liczbę -5 (przeciwna do +5)

1. Zapisać liczbę (+5);
2. Zamienić wszystkie 1/0 i 0/1;
3. Dodać 1.

00000101=-0+5=5 **(1)**

11111010 **(2)**

+ 1 **(3)**

11111011=-128+64+32+16+8+2+1=-5

Liczby rzeczywiste – zapis stałoprzecinkowy

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.

Liczby rzeczywiste – zapis stałoprzecinkowy

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).

Liczby zmiennopozycyjne

5 973 600 000 000 000 000 000 000 kg

(Masa Ziemii)

$5{,}9736 * 10^{24}$

5,9736 E+24

mantysa (precyzja)

cecha (wykładnik)

Notacja naukowa pozwala na kodowanie bardzo dużych / małych liczb

Liczby zmiennopozycyjne

$101{,}101$

$5{,}625_{10}$

$0{,}101101 * 2^3$

Mantysa znormalizowana dla liczb binarnych należy do przedziału $<\frac{1}{2},1)$.

W praktyce oznacza to, że przecinek należy ustawić w taki sposób, aby liczba miała postać:

0,1xxxxxx...

Dzięki normalizacji zapis staje się jednoznaczny.

Liczby zmiennopozycyjne

znak	cecha / wykładnik	mantysa / precyzja
1	**10010**	**1010110101**

Standard IEEE 754

pojedyncza precyzja	1	8	23	(32 bity)
podwójna precyzja	1	11	52	(64 bity)

Liczby zmiennopozycyjne

Przykład

10001101101110101

1 00011 01101110101

$$- 0{,}110\ 110101 * 2^{3} =$$

$$= 110{,}110101 = 6{,}828125_{10}$$

Część 3

Reprezentacja danych w komputerze