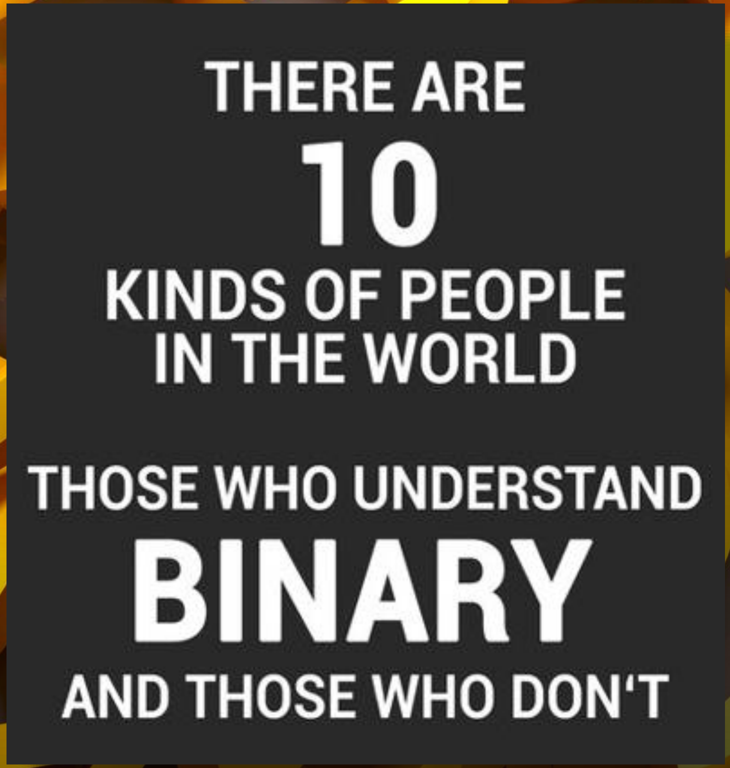
OSNOVE DIGITALNE ELEKTRONIKE

Brojevni sustavi

Zdravko Kunić
zdravko.kunic@racunarstvo.hr



Brojevni sustavi

Ishod 1	
	Konvertirati brojeve između brojevnih sustava
	Definirati brojevne sustave i opće principe digitalnog kodiranja

Sadržaj predavanja

- Prikaz podataka u digitalnom obliku
- Brojevni sustavi
- Binarni prikaz cijelih i razlomljenih brojeva
- Izbor optimalnog brojevnog sustava
- Pretvorba brojeva između brojevnih sustava
- Prikaz relativnih brojeva (brojeva s predznakom)
- Zapis s pomičnim zarezom - **IEEE754**

Prikaz podataka u digitalnom obliku

- Binarni vektor (niz bitova):
  - Npr. 10110100
- Niz bitova može imati različito značenje:
  - **broj**: prirodni, cijeli, realni, …
  - **znak/simbol**: slovo, znamenka, točka, zagrada, matematički simbol (znakovni kôd)
  - **specijalni znak**: upravljački, instrukcija, …

  - posebno značenje: logička varijabla

Zapis podataka u digitalnom obliku

- Značenje binarnog vektora nije apriori poznato
- Način zapisa podataka naziva se **format**
  - Svaka vrsta podatka (broj, znak…) zapisuje se u utvrđenom obliku
  - Formatom se utvrđuje organizacija i značenje pojedinih bitova

- Najjednostavniji način zapisa: zapis prirodnih binarnih brojeva
  - Svaki drugi tip podataka mora biti zapisan u obliku **kôda**
- **Kôd** = pridruživanje nekog značenja binarnom vektoru

Brojevni sustavi

- **Nepozicijski**
  - Pozicija znamenke ne određuje njezino značenje
  - Primjer: sustav rimskih brojeva

$$\boldsymbol{MMXXIII} \Rightarrow 2000\ (MM) + \ 20(XX) + 3(III) \ = 2023$$

  - Nedostatci:
    - za zapisivanje većih brojeva treba uvoditi nove znamenke
    - obavljanje aritmetičkih operacija je vrlo složeno
- **Pozicijski**
  - Pozicija znamenke određuje njezinu težinu
  - Težina predstavlja potenciju baze brojevnog sustava (faktor kojim se znamenka množi)
  - Primjer: dekadski sustav (baza je 10)

$$\mathbf{2023} = \mathbf{2} \cdot 10^3 + \mathbf{0} \cdot 10^2 + \mathbf{2} \cdot 10^1 + \mathbf{3} \cdot 10^0$$

Primjeri brojevnih sustava (1/2)

baza B	brojevni sustav	znamenke sustava (B)
2	binarni	0,1
3	ternarni	0,1,2
8	oktalni	0,1,2,3,4,5,6,7
10	dekadski	0,1,2,3,4,5,6,7,8,9
16	heksadekadski	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Primjeri brojevnih sustava (2/2)

dekadski	binarni	oktalni	heksadekadski
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Baza sustava

Općenito, broj možemo prikazati u obliku polinoma u kojem **baza sustava** može biti bilo koji **cijeli** broj B:

$$N_B = a_{n-1} \cdot B^{n-1} + a_{n-2} \cdot B^{n-2} + \cdots + a_1 \cdot B^1 + a_0 \cdot B^0$$
$$= \sum_{i=0}^{n-1} a_i \cdot B^i$$

B: *baza* ili *korijen* brojevnog sustava

a_i: **koeficijent** uz *i*-tu potenciju (težinu); a_i = {0, 1, …., B-1}, i = 0, 1, …., n-1 $\Rightarrow$ **znamenke**

Zadatak:

Prikažite broj **$(42)_8$** u obliku polinoma

$$N_B = a_{n-1} \cdot B^{n-1} + a_{n-2} \cdot B^{n-2} + \cdots + a_1 \cdot B^1 + a_0 \cdot B^0$$

$$= \sum_{i=0}^{n-1} a_i \cdot B^i$$

$N=$ **42** , $B=$ **8**

$$(\overset{1}{4}\overset{0}{2})_8 = 4 \cdot 8^1 + 2 \cdot 8^0$$

ALGEBRA

Prikaz razlomljenih brojeva

- Potencije baze koje odgovaraju znamenkama iza zareza su **negativne**
- Posebno se pretvara cjelobrojni a posebno razlomljeni dio broja

Zadatak:

Prikažite broj $\mathbf{(12{,}07)_8}$ u obliku polinoma

$$\overset{1}{1}\overset{0}{2},\overset{-1}{0}\overset{-2}{7}$$

$(12{,}07)_8 =$

$= 1 \cdot 8^{1} + 2 \cdot 8^{0} + 0 \cdot 8^{-1} + 7 \cdot 8^{-2}$

Pretvorba brojeva u različitim sustavima

Pretvorba **cijelog dekadskog** broja u neki drugi sustav:

- sukcesivno dijeljenje s bazom tog sustava
- **ostaci** dijeljenja s bazom predstavljaju **znamenke** konvertiranog broja
  - ostatak **prvog** dijeljenja predstavlja **najmanje značajnu** znamenku

Zadatak

$345_{10} = ?_2$

	ostatak
$345 : 2 = 172$	1
$172 : 2 = 86$	0
$86 : 2 = 43$	0
$43 : 2 = 21$	1
$21 : 2 = 10$	1
$10 : 2 = 5$	0
$5 : 2 = 2$	1
$2 : 2 = 1$	0
$1 : 2 = 0$	1

$$345_{10} = 101011001_2$$

Zadatak $345_{10} = ?_{16}$

ostatak

$$345 : 16 = 21 \quad 9$$
$$21 : 16 = 1 \quad 5$$
$$1 : 16 = 0 \quad 1$$

$$345_{10} = 159_{16}$$

Pretvorba u dekadski sustav

Izravno:

- odrediti dekadski zapis svake potencije baze izvornog sustava
- pomnožiti vrijednost svake znamenke s odgovarajućom težinom
- zbrojiti umnoške

Zadatak

Pretvorite binarni broj **10010,101** u dekadski sustav

$$
\begin{array}{l}
\overset{4\;3\;2\;1\;0\;-1\;-2\;-3}{\mathbf{10010{,}101}_2} = 1\cdot 2^4 + 0\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 0\cdot 2^0 \\
\qquad + 1\cdot 2^{-1} + 0\cdot 2^{-2} + 1\cdot 2^{-3} \\
\qquad = 1\cdot 16 + 1\cdot 2 + 1\cdot 0{,}5 + 1\cdot 0{,}125 \\
\qquad = 18{,}625
\end{array}
$$

$$\mathbf{10010{,}101_2 = 18{,}625_{10}}$$

ALGEBRA

Usporedba brojevnih sustava

Povećanjem baze sustava smanjuje se broj brojnih mjesta (znamenaka)

Baza sustava	Broj 11_{10}
2	1011
3	102
8	13
10	11
16	B

ALGEBRA

Optimalan brojevni sustav?

- Za prikaz nekog n-znamenkastog broja u sustavu s bazom B potrebno je n sklopova s B diskretnih stanja
- Ukupan broj različitih diskretnih stanja (v) računamo formulom:

$$v = B \cdot n$$

- Broj različitih brojeva N koji se mogu prikazati s n znamenaka:

$$N = B^n$$

- Primjer: 3-znamenkastim brojevima u sustavu s bazom 2 moguće je prikazati 8 različitih brojeva.

$$B=2,\ n=3 \quad \rightarrow \quad v = 2 \cdot 3 = 6,\ \ N = 2^3 = 8$$

Primjer usporedbe brojevnih sustava

Usporedba binarnog i dekadskog brojevnog sustava za $0 \le N \le 99$

N	Binarni:	Dekadski:
0	$0 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0$	$0 \cdot 10^1 + 0 \cdot 10^0$
1	$0 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$	$0 \cdot 10^1 + 1 \cdot 10^0$
2	$0 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$	$0 \cdot 10^1 + 2 \cdot 10^0$
...		
98	$1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$	$9 \cdot 10^1 + 8 \cdot 10^0$
99	$1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$	$9 \cdot 10^1 + 9 \cdot 10^0$

$n=7,\ B=2$ $\qquad$ $n=2,\ B=10$

$\mathbf{v_2} = 7 \cdot 2 = \mathbf{14}$ $\qquad$ $\mathbf{v_2 < v_{10}}$ $\qquad$ $\mathbf{v_{10}} = 2 \cdot 10 = \mathbf{20}$

Izbor optimalnog brojevnog sustava

$$N = B^n \text{ ; ln } \rightarrow \ln N = n \cdot \ln B \rightarrow n = \frac{\ln N}{\ln B}$$

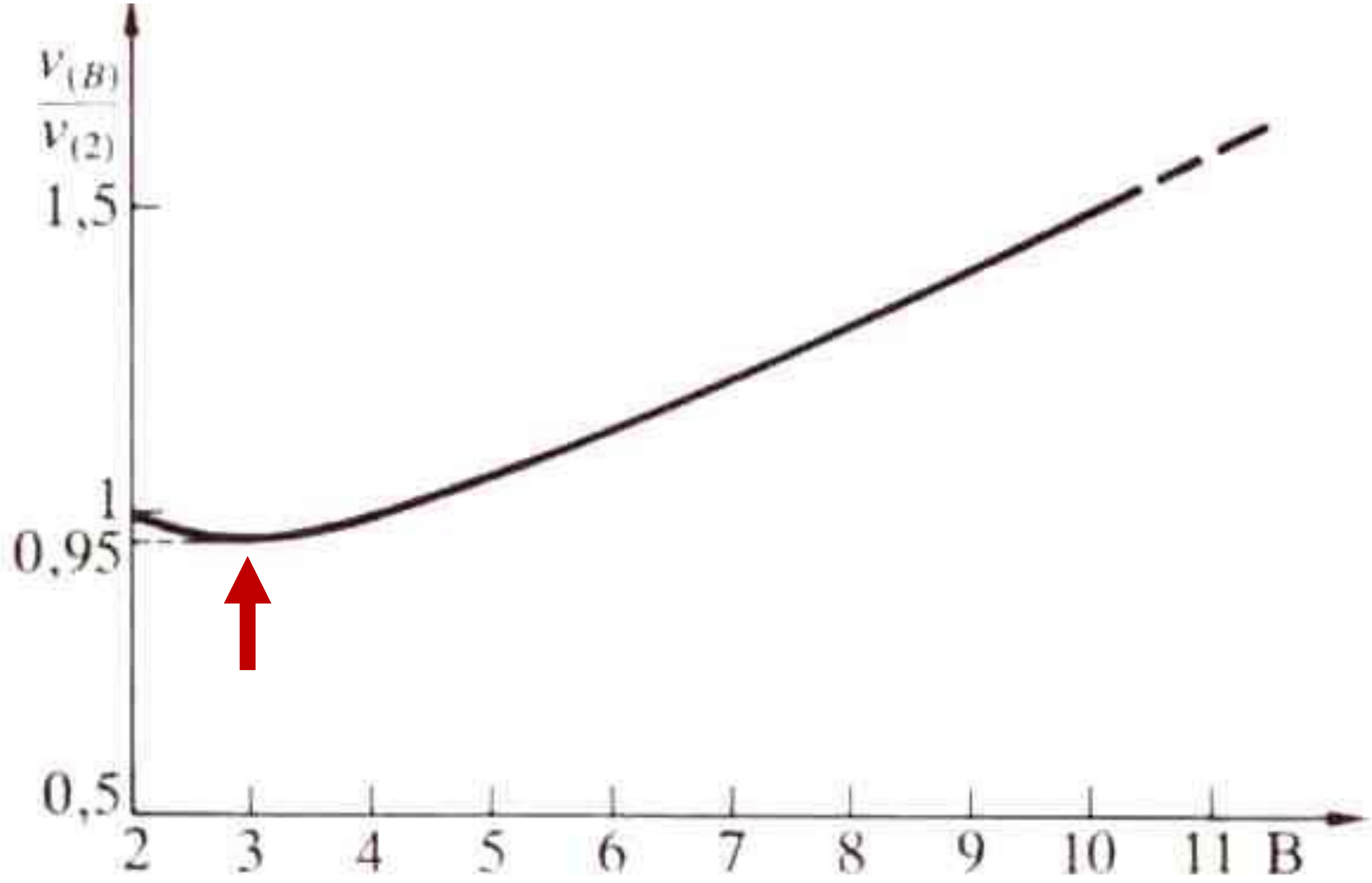
$$v = n \cdot B \rightarrow v = \frac{\ln N}{\ln B} \cdot B \text{ ; } \ln B \neq 0$$

Budući da je v ukupan broj diskretnih stanja koja treba fizički realizirati, optimalno je rješenje broj prikazati sa što manje ukupnih stanja. Stoga deriviramo izraz za v i izjednačavamo ga s nulom, čime dobivamo uvjet za minimum:

$$\frac{dv}{dB} = \ln N \cdot \frac{\ln B - 1}{(\ln B)^2} = 0 \rightarrow \ln B - 1 = 0 \rightarrow \boldsymbol{B = e = 2.7183 \ldots}$$

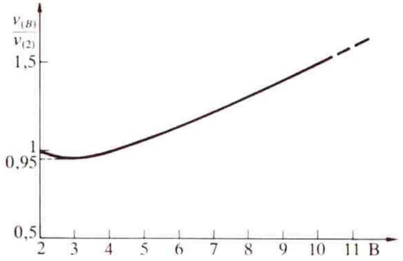
Izbor optimalnog brojevnog sustava

Optimalna baza brojevnog sustava = 3



Optimalna baza brojevnog sustava

- "najekonomičnija" baza: B = 3
  ➔ ternarni brojevni sustav:
  najbliži teorijskom minimumu



Zašto onda računala koriste binarni brojevni sustav?

- binarni brojevni sustav:
  ➔ lakša realizacija:
  tehnički bolji,
  uz samo 5% razlike od ternarnog

Oktalni i heksadekadski sustavi

- **pozicijski brojevni sustavi**, baza **8** odnosno **16**
- baza je **potencija broja 2** $\Rightarrow$ jednostavna pretvorba u binarni sustav
- veća baza $\Rightarrow$ manji broj znamenaka

Oktalni sustav

- baza sustava je **8**, znamenke su **0-7**
- svaka znamenka je predstavljena s **3 bita**
- jednostavna pretvorba

```
0    000
1    001
2    010
3    011
4    100
5    101
6    110
7    111
```

Primjeri

- **$101111011001100_2 = ?_8$**

101 111 011 001 100

5 7 3 1 4

$101111011001100_2 = \mathbf{57314_8}$

- **$765432_8 = ?_2$**

7 6 5 4 3 2

111 110 101 100 011 010

$765432_8 = \mathbf{111110101100011010_2}$

Heksadekadski sustav

- baza sustava je **16**, znamenke su **0-9**, **A-F**
- svaka znamenka je predstavljena s **4 bita**
- jednostavna pretvorba
- vrlo raširen brojevni sustav kao sažeti zapis binarnog
- 2 heksa znamenke ~ 1 oktet (1B)

```
0     0000       A     1010
1     0001       B     1011
   ...           C     1100
7     0111       D     1101
8     1000       E     1110
9     1001       F     1111
```

Primjeri

- **$1011110001110011100_2 = ?_{16}$**

0101 1110 0011 1001 1100

5 E 3 9 C

$01011110001110011100_2 =$ **$5E39C_{16}$**

- **$76A4C2_{16} = ?_2$**

7 6 A 4 C 2

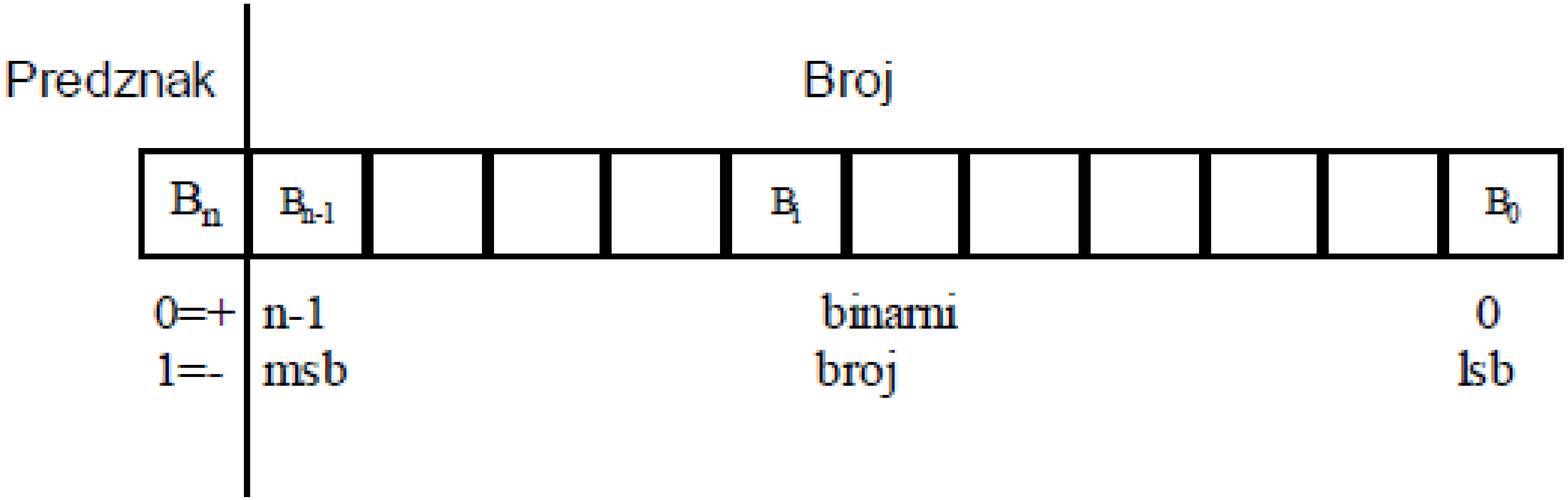
0111 0110 1010 0100 1100 0010

$76A4C2_{16} =$ **$011101101010010011000010_2$**

Prikaz relativnih brojeva

- Za prikaz pozitivnih i negativnih brojeva potreban je dodatni bit za predznak
  - Pozitivni broj 0
  - Negativni broj 1
- Ostali bitovi predstavljaju vrijednost

Brojevi s predznakom



Tri načina prikaza negativnih brojeva

a) Predznakom i brojem (veličinom, vrijednošću)
- Primjer: `+72 = 01001000, -72 = 11001000`

b) Predznakom i 1-komplementom broja
- 1. komplement odredimo tako da pretvorimo 0→1 i 1→0
- Primjer: `+72 = 01001000, -72 = 10110111`

c) 2-komplementom broja
- 2. komplement odredimo tako da 1. komplementu pribrojimo 1
- Primjer: `+72 = 01001000, -72 = 10111000`

a) Prikaz negativnih brojeva predznakom i brojem

- Negativni predznak se označava početnim bitom 1
- Pozitivni predznak se označava početnim bitom 0
- Primjeri:
  - `+72 = 01001000` `-72 = 11001000`
  - `+127 = 01111111` `-127 = 11111111`

- Raspon [-127,127], dvije nule: `00000000 i 10000000`

b) Prikaz negativnih brojeva predznakom i 1-komplementom

- 1-komplement odredimo tako da pretvorimo bitove 0→1 i 1→0
- Primjeri:
  - `+72 = 01001000` `-72 = 10110111`
  - `+127 = 01111111` `-127 = 10000000`
- Raspon [-127,127], dvije nule: `00000000 i 11111111`

c) Prikaz negativnih brojeva 2-komplementom

- 2-komplement odredimo tako da pretvorimo 0→1 i 1→0 te dobivenom broju pribrojimo 1
- Primjeri:

```
 +72 = 01001000 →  10110111
                 +        1
            -72 =  10111000

+127 = 01111111 →  10000000
                 +        1
           -127 =  10000001
```

Raspon [-128,127]

```
-128 = 10000000
```

Zapis s pomičnim zarezom - IEEE754

- Zapis po standardu **IEEE754** (32-bita) se sastoji od tri dijela:
  - **Predznak** (1 bit)
    - za pozitivan broj = 0, za negativan broj = 1
  - **Karakteristika** (8 bitova)
    - računa se kao *eksponent + 127;* valjani raspon: [-126,127]
    - Ostale vrijednosti su rezervirane za posebne indikatore (npr. beskonačnost)
  - **Decimalni dio mantise** (23 bita)
    - prva znamenka cijelog dijela broja je uvijek 1
    - bitovi decimalnog dijela mantise dopunjuju nulama do dužine od 23 bita

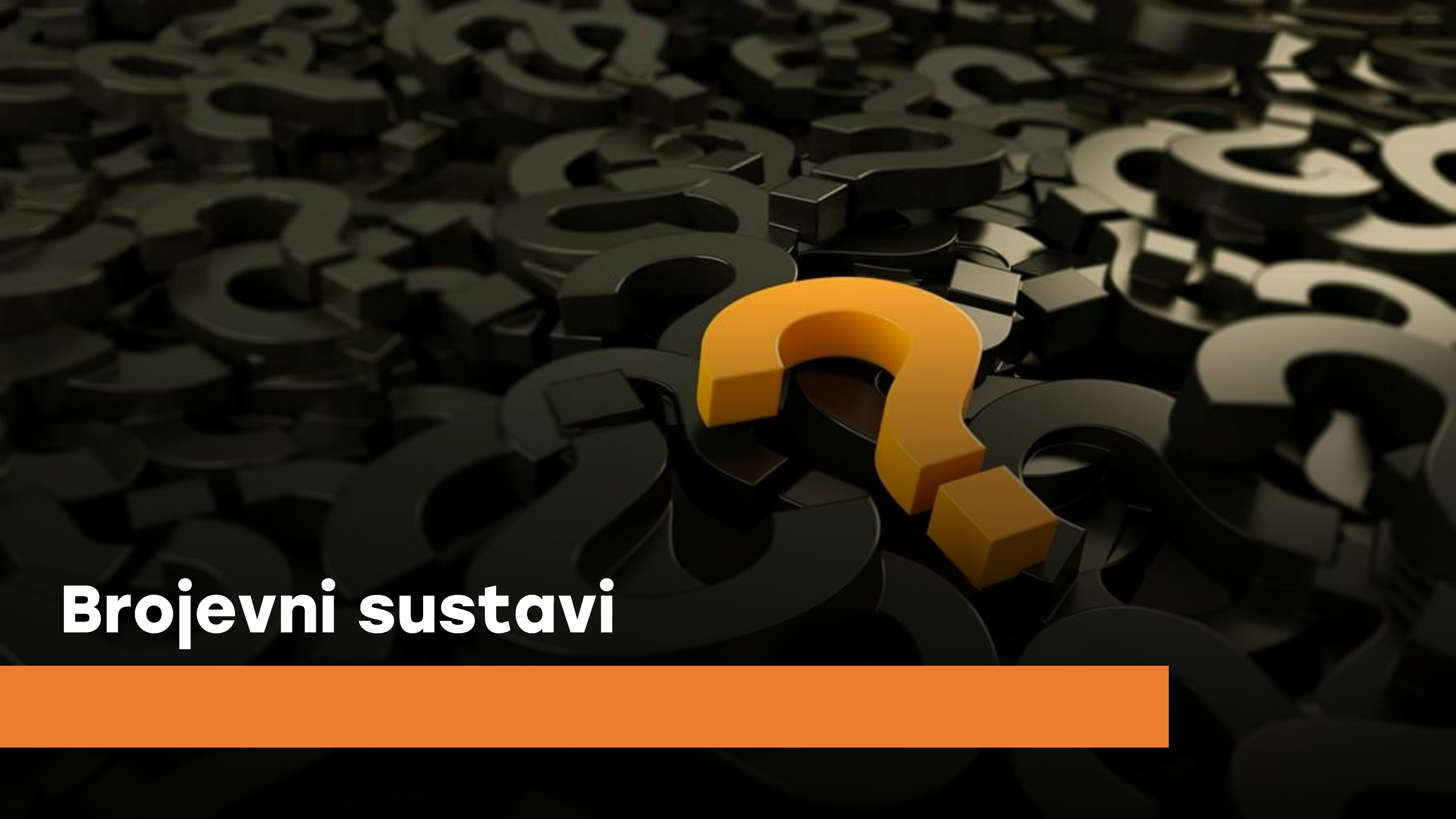
Pomoć za vježbu i razumijevanje: https://www.binaryconvert.com/convert_float.html

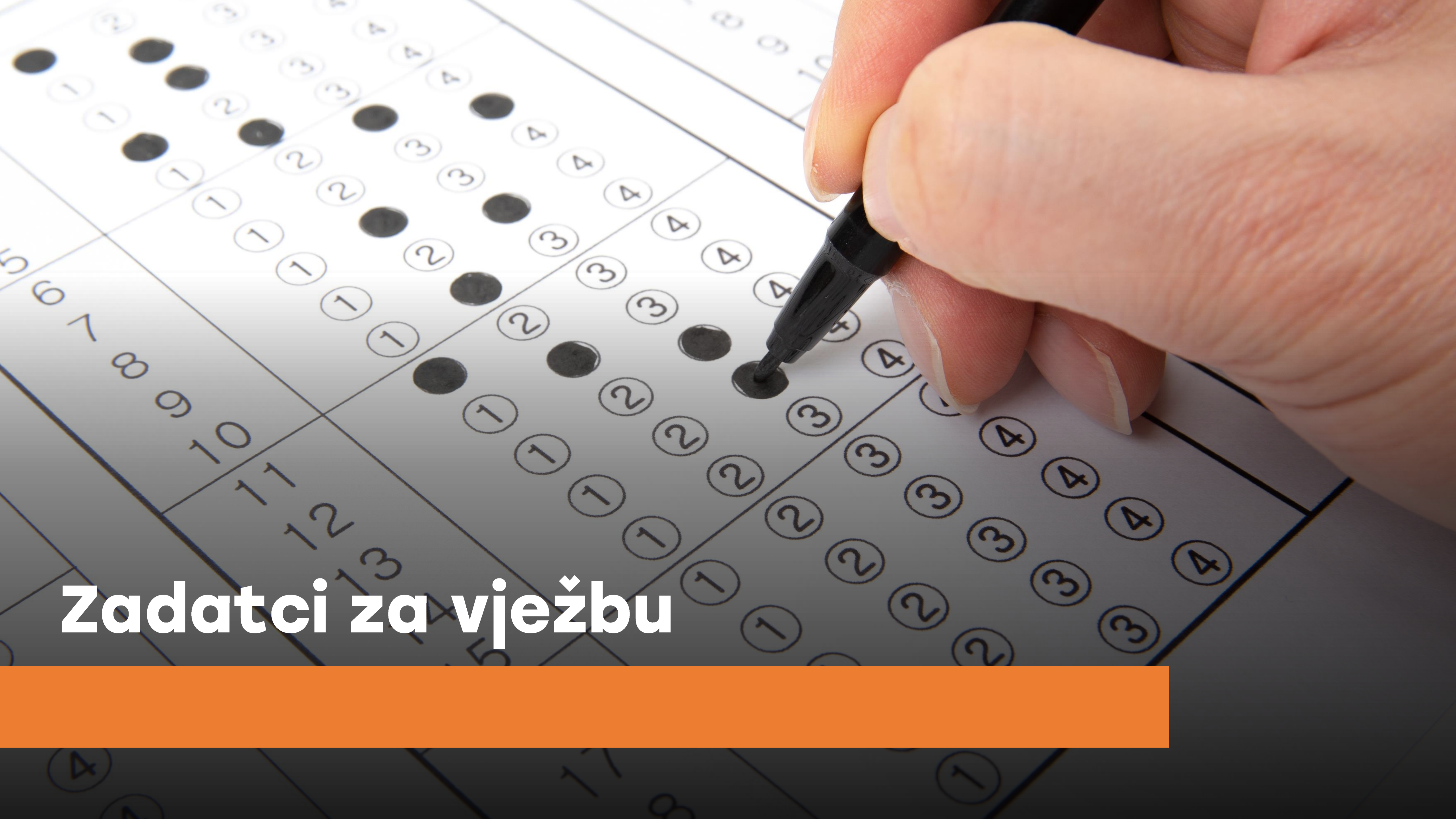
Primjer: $-125{,}125_{10} = ?$ (IEEE754, 32-bita)

1	10000001	11110100100000000000000
Predznak	Karakteristika (8 bita)	Decimalni dio mantise (23 bita)
-125,125 Broj je negativan pa je predznak 1	$125{,}125 = 1111101{,}001$ $= 1{,}111101001 \cdot 2^6$ → eksponent = 6 eksponent + 127 = $= 6+127=133_{10}$ $133_{10} = 10000101_2$	$125{,}125 = 1{,}111101001 \cdot 2^6$ decimalni dio mantise = 111101001 (ostali bitovi su 0)

11000010111110100100000000000000

Brojevni sustavi





Zadatci za vježbu

Podsjetnik za pretvorbu dekadskih brojeva

- **Cijeli dekadski broj** se pretvara u drugu bazu postupkom uzastopnog **dijeljenja** s novom bazom
- **Decimalni ostatak** se pretvara u drugu bazu postupkom uzastopnog **množenja** s novom bazom
  - Cjelobrojni dio rezultata množenja predstavlja binarnu znamenku (0 ili 1)
  - U nastavak množenja ulazi samo decimalni dio

Primjer: $14_{10} = ?_2$

14 : 2 = 7 ostatak 0 (najmanje važna znamenka)

7 : 2 = 3 ostatak 1

3 : 2 = 1 ostatak 1

1 : 2 = 0 ostatak 1

$$14_{10} = 1110_2$$

Primjer: $18{,}296875_{10} = ?_2$

- $18 : 2 = 9$ 0
- $9 : 2 = 4$ 1
- $4 : 2 = 2$ 0
- $2 : 2 = 1$ 0
- $1 : 2 = 0$ 1

- $0{,}296875 \cdot 2 = 0{,}59375$ 0
- $0{,}59375 \cdot 2 = 1{,}1875$ 1
- $0{,}1875 \cdot 2 = 0{,}375$ 0
- $0{,}375 \cdot 2 = 0{,}75$ 0
- $0{,}75 \cdot 2 = 1{,}5$ 1
- $0{,}5 \cdot 2 = 1$ 1

$$18{,}296875_{10} = 10010{,}010011_2$$

ALGEBRA

Pretvorite dekadski broj 217 u heksadekadski

- 217 : 16 = 13   ostatak   9
- 13 : 16 = 0   ostatak 13

$$217_{10} = D9_{16}$$

Pretvorite binarne brojeve u oktalne i heksadekadske:

- 11010111
- 1001101,011
- 111000010101101

Rješenje

Pretvorba binarnog broja u oktalni

$11010111_2 \sim 011\ 010\ 111$

3 2 7 = 327_8

Pretvorba binarnog broja u heksadekadski

$11010111_2 \sim 1101\ \ 0111$

(13)D 7 = $D7_{16}$

Pretvorite oktalne brojeve u binarne:

- 737
- 1242
- 23

Rješenje

737 ~ 7 3 7

111 011 111

$$(737)_8 = (111011111)_2$$

Pretvorite heksadekadske brojeve u binarne:

- 2FA
- 12C
- 4B3

Rješenje

2FA ~ 2 F A

0010 1111 1010

$$(2FA)_{16}=(1011111010)_2$$

Pretvorite binarne brojeve u pripadne 1- i 2-komplemente

- 1011
- 11010

Prikažite brojeve u obliku polinoma

- $(101011)_2$
- $(42)_8$
- $(36)_{10}$
- $(11,31)_{10}$
- $(F8)_{16}$
- $(1101,11)_2$
- $(12,07)_8$
- $(38,D2)_{16}$

Pretvorite dekadske brojeve u binarne ekvivalente:

- 14
- 64
- 1024
- 17,324
- 18,296875

Zadatak za vježbu

- U 32-bitnom registru zapisan je broj prema IEEE754 standardu.
- Heksadekadski ekvivalent zapisa je **C13E0000**

Koji **dekadski** broj je zapisan u registru?

Rješenje

$C13E0000_{16}$ = 1 10000010 011 1110 0000 0000 0000 $0000_{2\ (IEEE754)}$

- predznak = 1
- karakteristika = $10000010_2 = 130_{10}$
  - eksponent = karakteristika -127 = 130 – 127 = 3
- mantisa = $1{,}011111 \cdot 2^3$
  - prvi broj je uvijek 1
  - možemo zanemariti sve nule zdesna jer ne nose dodatnu informaciju
  - pomaknemo zarez za 3 mjesta udesno: 1,011111 → 1011,111
  - pretvorimo broj $1011{,}111_2$ u dekadski:
    - $(1011.111)_2 = (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (1 \times 2^{-2}) + (1 \times 2^{-3}) = (11.875)_{10}$
  - dodamo predznak (**1**)

Rezultat je $\mathbf{-11{,}875_{10}}$

LITERATURA:

- Peruško, Glavinić: **Digitalni sustavi**
  - str. 31 – 42
- Mano, Kime, Martin: **Logic and Computer Design Fundamentals**
  - Chapter 1-3 Number Systems, str. 31 – 40
  - Chapter 9-7 Floating point computations, pages 525-530