



1. Brojni sistemi

1. Ako se za prikaz binarnog broja koristi razvijen eksponencijalni zapis, broj se lako prevodi u dekadni brojni sistem kao u sledećem primeru:

$$\begin{aligned}
1110_{(2)} &= 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 \\
&= 8 + 4 + 2 + 0 \\
&= 14_{(10)}
\end{aligned}$$

Korišćenjem navedene procedure prevesti u dekadni brojni sistem sledeće binarne brojeve:

- | | | | |
|------------------|-------|---------------------|-------|
| a) $1100_{(2)}$ | _____ | d) $10101101_{(2)}$ | _____ |
| b) $101_{(2)}$ | _____ | e) $11100111_{(2)}$ | _____ |
| c) $10001_{(2)}$ | _____ | f) $10111101_{(2)}$ | _____ |

2. Kompletirajte sledeće zapise:

- a) _____₍₂₎ = $_ * _ + _ * _ + _ * _ + _ * _ + _ * _$
= 16 + 0 + 4 + 0 + 1
= 21₍₁₀₎
- b) $110000_{(2)}$ = $_ * _ + _ * _ + _ * _ + _ * _ + _ * _ + _ * _$
= _____ + _____ + _____ + _____ + _____ + _____
= _____₍₁₀₎
- c) _____₍₂₎ = $1 * 2^5 + 0 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$
= _____ + _____ + _____ + _____ + _____ + _____
= _____₍₁₀₎

3. Zabeležite (kodirajte) sledeće binarne brojeve u oktalnom brojnom sistemu:

- | | | | |
|-----------------|-------|-------------------|-------|
| a) 110110011 | _____ | e) 11010 | _____ |
| b) 1010101 | _____ | f) 100000001001 | _____ |
| c) 1110001101 | _____ | g) 101011011111 | _____ |
| d) 1010 | _____ | h) 10111011 | _____ |

4. Zabeležite (kodirajte) sledeće oktalne brojeve u binarnom brojnom sistemu:

- | | | | |
|---------|-------|---------|-------|
| a) 476 | _____ | e) 1001 | _____ |
| b) 1045 | _____ | f) 456 | _____ |
| c) 201 | _____ | g) 1327 | _____ |
| d) 3321 | _____ | h) 717 | _____ |

5. Kompletirajte sledeće zapise:

- a) _____₍₈₎ = $_ * _ + _ * _ + _ * _$
= 192 + 56 + 0
= _____₍₁₀₎
- b) $1011_{(8)}$ = $_ * _ + _ * _ + _ * _ + _ * _$
= _____ + _____ + _____ + _____
= _____₍₁₀₎

6. Zabeležite (kodirajte) sledeće binarne brojeve u heksadekadnom brojnom sistemu:

- | | |
|---------------------|-----------------------|
| a) 110110011 _____ | e) 11010 _____ |
| b) 1010101 _____ | f) 100000001001 _____ |
| c) 1110001101 _____ | g) 101011011111 _____ |
| d) 1010 _____ | h) 10111011 _____ |

7. Zabeležite (kodirajte) sledeće heksadekadne brojeve u binarnom brojnom sistemu:

- | | |
|---------------|----------------|
| a) A12 _____ | e) DE161 _____ |
| b) CAB _____ | f) 40F0B _____ |
| c) 4F03 _____ | g) 1010 _____ |
| d) FED _____ | h) 777 _____ |

8. Kompletirajte sledeće zapise:

- a) $3F1_{(16)} = _ * _ + _ * _ + _ * _$
 $= _ + _ + _$
 $= __{(10)}$
- b) $_____{(16)} = _ * _ + _ * _ + _ * _ + _ * _ + _ * _ + _ * _$
 $= _ + _ + _ + _ + _ + _$
 $= 313_{(10)}$
- c) $_________{(16)} = 7 * 16^3 + 0 * 16^2 + 12 * 16^1 + 15 * 16^0$
 $= _ + _ + _ + _$
 $= _____{(10)}$

9. Zapišite 8-bitni zapis (ili 8 najmlađih binarnih cifara iz binarnog zapisa) i odgovarajući heksadekadni ekvivalent sledećih dekadnih brojeva:

- | | |
|--------------|---------------|
| a) 127 _____ | e) 256 _____ |
| b) 128 _____ | f) 904 _____ |
| c) 222 _____ | g) 1010 _____ |
| d) 255 _____ | h) 1072 _____ |

10*. Napisati program CONVINT koji za zadati ceo broj (≤ 65535 dekadno) i osnovu brojnog sistema (≤ 16) vrši prevođenje tog broja u dekadni brojni sistem, odnosno iz dekadnog u brojni sistem zadate osnove, u zavisnosti od zahtevane operacije (*u dekadni* ili *iz dekadnog*). Ulazni podaci su: osnova brojnog sistema, broj koji se prevodi i vrsta prevođenja. Program treba da spreči zadavanje nekorektnih ulaznih podataka.

11*. Napisati program CONVRE koji za zadat realni broj u obliku fiksnog zareza (znak, cifre celog dela, tačka, cifre razlomljenog dela) i osnovu brojnog sistema (≤ 16) vrši prevođenje tog broja u dekadni brojni sistem, odnosno iz dekadnog u brojni sistem zadate osnove, u zavisnosti od zahtevane operacije (*u dekadni* ili *iz dekadnog*). Broj može da ima najviše 32 cifre u binarnom zapisu. Program treba da spreči zadavanje nekorektnih ulaznih podataka.



2. Aritmetičke operacije

1. Izvršiti sledeća sabiranja u binarnom brojnem sistemu:

$$\begin{array}{r}
 \text{a) } 11101 \\
 + 1001 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{b) } 10011 \\
 + 1111 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{c) } 110111 \\
 + 110111 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{d) } 1010101 \\
 + 101011 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{e) } 1011 \\
 0110 \\
 1110 \\
 + 0101 \\
 \hline
 \end{array}$$

2. Smatrajući da su sledeći binarni brojevi zabeleženi u 8-bitnim registrima izvršiti tražena sabiranja i uz svaki zbir konstatovati da li je "stao" u registar (CF=0 - carry fleg) ili nije (CF=1).

$$\begin{array}{r}
 \text{a) } 10110111 \\
 + 00011110 \\
 \hline
 \text{CF=}
 \end{array}
 \quad
 \begin{array}{r}
 \text{b) } 10011000 \\
 + 01100110 \\
 \hline
 \text{CF=}
 \end{array}
 \quad
 \begin{array}{r}
 \text{c) } 11011001 \\
 + 11000111 \\
 \hline
 \text{CF=}
 \end{array}
 \quad
 \begin{array}{r}
 \text{d) } 10101010 \\
 + 01010111 \\
 \hline
 \text{CF=}
 \end{array}$$

3. Korišćenjem potpunog komplementa izvršiti sledeća binarna oduzimanja

$$\begin{array}{r}
 \text{a) } 10110111 \\
 - 00011110 \\
 \hline
 10110111 \\
 + 11100010 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{b) } 10011000 \\
 - 01100110 \\
 \hline
 10011000 \\
 + 10011010 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{c) } 11011001 \\
 - 01000111 \\
 \hline
 11011001 \\
 + 10111001 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{d) } 10101010 \\
 - 01010111 \\
 \hline
 10101010 \\
 + 10101010 \\
 \hline
 \end{array}$$

4. Izvršiti sledeća množenja u binarnom brojnem sistemu:

$$\begin{array}{ll}
 \text{a) } 11101 * 101 & \text{c) } 110111 * 110 \\
 \text{b) } 10011 * 1011 & \text{d) } 1010101 * 1001
 \end{array}$$

5. Izvršiti sledeća sabiranja u oktalnom brojnem sistemu:

$$\begin{array}{r}
 \text{a) } 1340 \\
 + 207 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{b) } 6402 \\
 + 435 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{c) } 1326 \\
 + 5507 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{d) } 63510 \\
 + 7741 \\
 \hline
 \end{array}
 \quad
 \begin{array}{r}
 \text{e) } 532401 \\
 + 65204 \\
 \hline
 \end{array}$$

6. Izvršiti sledeća množenja u oktalnom brojnem sistemu:

$$\begin{array}{ll}
 \text{a) } 137 * 20 & \text{c) } 1701 * 45 \\
 \text{b) } 1234 * 56 & \text{d) } 10765 * 143
 \end{array}$$

7. Izvršiti sledeća sabiranja u heksadekadnom brojnem sistemu:

a) 1BEA b) DEED c) FEED d) 4795A e) 123456
 +2215 +3112 +CEED + B04B +ABCDEF
 ----- ----- ----- ----- -----

8. Izvršiti sledeća množenja u heksadekadnom brojnom sistemu:

a) 137*2A _____ c) F0F0*DAB _____
 b) 1AD*63 _____ d) C128*A09 _____

9. Izvršiti sledeća deljenja u brojnom sistemu zadate osnove N:

a) 11:11011 (N=2) _____ c) 45:7200 (N=8) _____
 b) 110:10100 (N=2) _____ d) E:C4 (N=16) _____

10. Kompletirajte sledeću tabelu:

	Dekadno	Binarno	Oktalno	Heksadekadno
1.	0.92			
2.		1101111.1011000		
3.				1.00C
4.			276.532	
5.	.25			
6.			3.33	

11*. Napisati program TABLICE koji za zadatu osnovu brojnog sistema (≤ 16) generiše tablice sabiranja i množenja. Program treba da spreči zadavanje nekorektnih ulaznih podataka.

12*. Napisati program KALKULATOR koji omogućava sabiranje, oduzimanje, množenje i deljenje brojeva u brojnom sistemu zadate osnove n (≤ 16). Ulazni podaci su: osnova brojnog sistema, argumenti i operacija. Program treba da spreči zadavanje nekorektnih ulaznih podataka.



3. BCD kodovi

1. Sledećom tabelom dati su neki od internih (BCD) kodova brojnih podataka.

	8421 NBCD	XS3 Stibitz-ov	2421 Aiken-ov	5421	84-2-1	753-6	ciklični1 Gray-ov	ciklični2
0	0000	0011	0000	0000	0000	0000	0000	0001
1	0001	0100	0001	0001	0111	1001	0001	0101
2	0010	0101	0010	0010	0110	0111	0011	0111
3	0011	0110	0011	0011	0101	0010	0010	1111
4	0100	0111	0100	0100	0100	1011	0110	1110
5	0101	1000	1011	1000	1011	0100	0111	1100
6	0110	1001	1100	1001	1010	1101	0101	1000
7	0111	1010	1101	1010	1001	1000	0100	1001
8	1000	1011	1110	1011	1000	0110	1100	1011
9	1001	1100	1111	1100	1111	1111	1000	0011

Kod je **komplementaran** ako su kodovi komplementarnih dekadnih cifara (čiji je zbir jednak najvećoj cifri, tj. 9) takvi da njihov zbir jednak 1111 - tj. najvećem binarnom broju koji se može zapisati sa 4 cifre. Od navedenih kodova komplementarni su:

Kod je **težinski** ako jedinici na svakoj poziciji uvek možemo pridružiti odgovarajući težinski ekvivalent. Na primer, kod prirodnog BCD kod (NBCD) jedinica na poziciji 20 uvek vredi 1, na poziciji 21 -2, 22 - 4 i 23 - 8, pa se ovaj kod zove i kod "8421". Od navedenih, težinski su:

Kod je **cikli-an** ako se kodovi uzastopnih cifara razlikuju samo na jednoj poziciji. Od navedenih kodova svojstvo cikličnosti imaju:

Kod je **paran** ako se kodovi svih parnih cifra završavaju istom binarnom cifrom, a svih neparnih onom drugom binarnom cifrom. Od navedenih svojstvo parnosti zadovoljavaju:

2. Napisati sledeće dekadne brojeve u NBCD kodu (8421):

- a) 7.77 _____ c) 313.8 _____
b) 12.139 _____ d) 101.59 _____

3. Predstaviti dekadni broj 847

- a) u NBCD kodu (8421) _____ c) u STIBITZOVOM kodu _____
b) u EJKENOVOM kodu (2421) _____ d) u GREJOVOM kodu _____

4. Zapisati sledeće NBCD brojeve u dekadnom brojnom sistemu:

- a) 1100110.00111 _____ c) 1100101.1100101 _____
b) 110100.001 _____ d) 10101.10101 _____

5. Zadati su brojevi 1000 0101 0001 1001 u NBCD kodu. Zapisati taj isti broj u:

- a) Ejkkenovom (2421) kodu _____ c) BCD kodu sa težinama $84(-2)(-1)$ _____
b) Stibitz-ovom "višak 3" kodu _____ d) BCD kodu sa težinama $753(-6)$ _____

6. Dat je niz binarnih cifara 0010010000110101. Dekodovati taj niz ako on predstavlja:

- a) dekadne cifre u NBCD kodu _____ b) dekadne cifre u Grejovom kodu _____

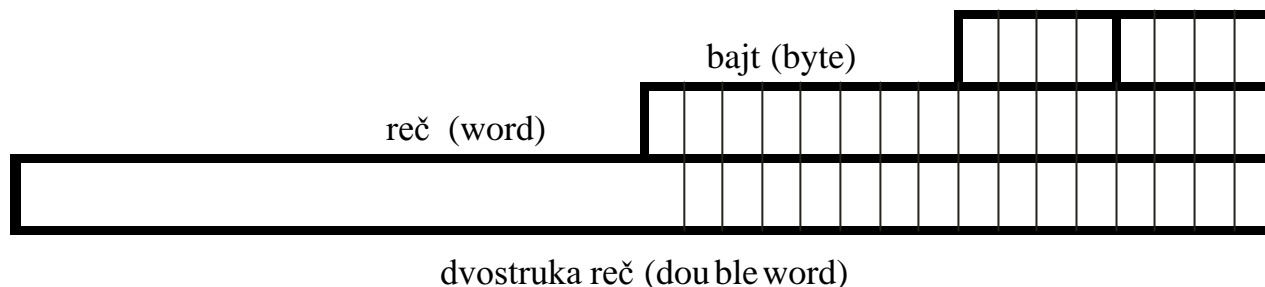
7*. Napisati program koji vrši konverziju NBCD brojeva u prave binarne brojeve korišćenjem *algoritma sa sabiranjem*. Prema tom algoritmu se prvo svakoj binarnoj cifri NBCD broja pridruži odgovarajući težinski faktor. Idući sa desna ulevo, težinski faktori su: 1, 2, 4, 8, 10, 20, 40, 80, 100 itd. Zatim se izvrši sabiranje binarnih ekvivalenata svih težinskih faktora koji odgovaraju bitovima u NBCD broju čija je vrednost jedinica. U ovu svrhu je zgodno koristiti prethodno formiranu tabelu binarnih ekvivalenata težinskih faktora iz koje se vidi da je npr. $80_{10}=101000_2$, $100_{10}=11001000_2$, $200_{10}=11001000_2$ itd.

8*. Napisati program koji vrši konverziju NBCD brojeva u prave binarne brojeve korišćenjem *algoritma pomeranja sa oduzimanjem*. Kod ovog algoritma NBCD broj se podeli u grupe od po 4 bita počevši od najmlađe pozicije. Izvrši se pomeranje za jedno mesto udesno zadržavajući granice dekada. Cifra NBCD broja koja izlazi van granice postaje najmlađa cifra binarnog prevoda. Zatim se ispituje da li su novi brojevi u granicama dekada veći od 8, i ako jesu od njih se oduzima 3. Pomeranje i ispitivanje se ponavlja sve dok se udesno ne pomere svi bitovi NBCD broja. Cifre koje izlaze na desnu stranu su cifre binarnog ekvivalenta NBCD broja. Za ilustraciju opisanog algoritma izvršimo konverziju dekadnog broja 13 u binarni ekvivalent.

13_{10}	0001 0011		
	000 1001 1	pomeraj	
	000 0110	oduzimanje 3	
	00 0011 0	pomeraj	
	0 0001 1	pomeraj	
	0000 1	pomeraj	1101_2



4. Registrovanje celih brojeva



1. Koji je redni broj najstarijeg (H.O) bita u bajtu? _____ A u reči? _____
Koliko u dvostrukoj reči? _____
2. Koliko bitova ima u bajtu (byte)? _____ A koliko bajtova u dvostrukoj reči? _____
3. Koji je redni broj najmlađeg (L.O) bita u starijem polubajtu - niblu (nibble) bajta? _____
A redni broj najstarijeg bita u mlađem niblu? _____
4. Koje karaktere u asemblerskim programima koristimo kao sufiks za oznaku heksadekadnih brojeva? _____ Binarni brojeva? _____ A dekadnih? _____
Ako se iza broja ne napiše sufiks, koji se brojni sistem podrazumeva? _____

POTPUNI KOMPLEMENT

Primer: Izračunajte potpuni komplement 16-bitne vrednosti F1Eh

- ♦ 1) Konvertujemo vrednost u 16-bitni binarni broj F1Eh = 0000 1111 0001 1110b
- ♦ 2) Invertujemo sve bitove . NOT 0000 1111 0001 1110 = 1111 0000 1110 0001
- ♦ 3) Dodamo jedan. 1111 0000 1110 0001 + 1 = 1111 0000 1110 0010
- ♦ 4) Vratimo nazad u heksadekadni zapis. -(F1Eh) = F0E2h

Primer: Izračunajte potpuni komplement 16-bitne vrednosti F0E2h:

- ♦ 1) Konvertujemo vrednost u 16-bitni binarni broj : F0E2h = 1111 0000 1110 0010b
- ♦ 2) Invertujemo sve bitove. NOT 1111 0000 1110 0010 = 0000 1111 0001 1101
- ♦ 3) Dodamo jedan . 0000 1111 0001 1101 + 1 = 0000 1111 0001 1110
- ♦ 4) Vratimo nazad u heksadekadni zapis. 0000 1111 0001 1110b = 0F1Eh.

Napomenimo da je: $-(F1Eh) = F1Eh$.

4. Koliko iznose najmanji i najveći dekadni broj koji se mogu zapisati

sa 8 bitova bez znaka _____ sa 8 bitova u PK _____

sa 16 bitova bez znaka _____ sa 16 bitova u PK _____

sa 32 bita bez znaka _____ sa 32 bita u PK _____



5. ASCII kodovi

Standardni ASCII set karaktera sadrži 128 različitih kodova koji reprezentuju različite vrste znakova. Većina računara koristi 8-bitnu reprezentaciju ASCII karaktera, pri čemu je H.O. bit postavljen na 0 (umesto izvornog 7-bitnog koda).

ASCII zapis se lako deli u 4 grupe korišćenjem bitova 5 i 6 na sledeći način:

Bit 6	Bit 5	Grupa karaktera
0	0	Kontrolni karakteri
0	1	Cifre & Interpunkcijski znaci
1	0	Velika slova & Specijalni znaci
1	1	Mala slova & Specijalni znaci

Treba imati na umu sledeće činjenice:

- ♦ Kod blanko znaka manji je od koda ma kog slova, cifre i uopšte, ma kog grafičkog simbola;
- ♦ Kodovi cifara uređeni su u rastući redosled i idu bez razmaka. Zato kada je zadovoljena nejednakost:

$$\text{kod('0')} \leq \text{kod(znak)} \leq \text{kod('9')}$$

znamo da je znak cifra i da važi:

$$\text{kod}(i) = \text{kod('0')} + i; \text{ gde je } 0 \leq i \leq 9.$$

Primetimo da je $\text{kod('0')} < 0$.

- ♦ Kodovi velikih slova latinice A..Z (26 slova) uređeni su saglasno abecedi i takođe idu bez razmaka. Zato kada je zadovoljena nejednakost:

$$\text{kod('A')} \leq \text{kod(znak)} \leq \text{kod('Z')}$$

znamo da je znak veliko slovo i da je kod i-tog velikog slova (pri numeraciji od 0) jednak zbiru $\text{kod('A')} + i$;

- ♦ Analogno tvrđenje važi za mala slova;

1. Dekodirajte sledeće poruke zabeležene u ASCII kodu:

a) 6C 75 78 75 72 79 20 63 61 72 _____

b) 41 4E 41 4C 59 53 49 53 _____

c) 31 39 38 39 31 39 39 30 31 39 39 31 _____

d) 53 61 73 74 61 6E 61 6B 20 6A 65 20 69 73 70 72 65 64 20 41 6C 62 61 6E 69 6A 65
20 75 20 73 72 65 64 75 20 75 20 32 33 2E 33 30 _____

2. Kojem dekadnom broju (ili slovu abecede - vodite računa o velikim i malim slovima) odgovara binarni sadržaj registra 00110111 ako se interpretira kao

a) pakovan BCD broj _____ b) označen broj _____

c) ASCII kod _____

3. Kojem dekadnom broju (ili slovu abecede - vodite računa o velikim i malim slovima) odgovara binarni sadržaj registra 01100001 ako se interpretira kao

a) pakovan BCD broj _____ b) označen broj _____

c) ASCII kod _____

4. Kojem dekadnom broju (ili slovu abecede - vodite računa o velikim i malim slovima) odgovara binarni sadržaj registra 01000001 ako se interpretira kao

a) pakovan BCD broj _____ b) označen broj _____

c) ASCII kod _____

LOGI^KE OPERACIJE

5. Izračunajte

FACEh AND F0F0h _____ FBh AND 54h _____

12h OR 34h _____ FBh OR 54h _____

5Ah XOR A5h _____ NOT 4Fh _____

6. Koju logičku operaciju treba izvršiti da bi se konvertovala mala slova u velika?
_____ Za konverziju velikih u mala slova? _____ Za alternativnu
promenu - velikih u mala, a malih u velika? _____

7*. Napisati funkcije koje za dati karakter vraćaju znak koji se dobija po sledećim pravilima:

a) UVELIKA za malo slovo vraća odgovarajuće veliko slovo, a drugi karaktere vraća bez promene;

b) UMALA koja konvertuje velika slova u mala;

c) UCONTRA koja konvertuje velika slova u mala, mala u velika, a ostale znake vraća bez promene;

d) SIFRA koja za svaki znak čiji je ASCII kod >11d vraća znak čiji je kod za 1 manji.

Zatim napisati program koji korišćenjem ovih funkcija konvertuje ulazni string u skladu sa pravilom koje izabere korisnik.