

Uvod u organizaciju računara

kolokvijum 2009, smerovi M,N,V,L

REŠENJA

1. a) Predstaviti sledeće brojeve u navedenim osnovama u zapisima znak i apsolutna vrednost, nepotpuni i potpuni komplement:

a) $(-4094)_{10} = ()_{16}^5$ b) $(1031)_{10} = ()_8^6$

- b) Zapise $(FFFABF)_{16}$ i $(01101)_7$ u potpunom komplementu prevesti u osnovu 10.

a)

a) $(-4094)_{10} = ()_{16}^5$

X_i	4094	255	15
y_i	E	F	F

smer čitanja ←

Prevod apsolutne vrednosti broja -4094 u heksadekadni sistem zapisan sa 5 cifara je 00EFF.

Traženi zapisi broja -4094 u polju širine 5 su:

Znak i apsolutna vrednost: **F0EFF**

Nepotpuni komplement: **FF001**

Potpuni komplement: **FF002**

b) $(1031)_{10} = ()_8^6$

X_i	1031	128	16	2
y_i	7	0	0	2

smer čitanja ←

Pošto je broj pozitivan, zapisi dužine 6 u obliku znaka i apsolutne vrednosti, nepotpunog i potpunog komplementa su: **002007**.

- b) $(FFFABF)_{16}$ - broj je negativan jer je cifra najveće težine jednaka najvećoj cifri heksadekadnog sistema. Vrednost broja se može izračunati preko tabele sa vrednostima heksadekadnih pozicija datog broja u potpunom komplementu:

3	2	1	0	heksadekadna pozicija
-4096	256	16	1	vrednost pozicije
F	A	B	F	cifre broja

Vrednost broja računa se kao zbir proizvoda vrednosti pozicije i vrednosti cifre na toj poziciji, pri čemu je vrednost cifre na poziciji najveće težine 0 ili -1, u zavisnosti od toga da li je broj pozitivan ili negativan.

Uz primenu pravila za upisivanje zapisa u potpunom komplementu u kraću reč, dobija se:
 $(FFFABF)_{16} = (FABF)_{16} = -16^3 + (A)_{16} \cdot 16^2 + (B)_{16} \cdot 16^1 + (F)_{16} \cdot 16^0 = -4096 + 2751 = -1345$

Napomena: dekadna vrednost broja $(FFFABF)_{16}$ može se izračunati i nalaženjem njegove apsolutne vrednosti. Apsolutna vrednost broja dobija se komplementiranjem vrednosti $(FFFABF)_{16}$ i jednaka je $(000541)_{16}$. Njena dekadna vrednost je: $(5)_{16} * 16^2 + (4)_{16} * 16^1 + (1)_{16} * 16^0 = 1345$. Dekadna vrednost broja $(FFFABF)_{16}$ je **-1345**.

$(01101)_7$ – broj je pozitivan jer je cifra najveće težine 0. Vrednost broja jednaka je:
 $(01101)_7 = (1)_7 * 7^3 + (1)_7 * 7^2 + (1)_7 * 7^0 = 343 + 50 = \mathbf{393}$

2. Prevesti u 8-bitne označene binarne brojeve i izvršiti množenje Butovim algoritmom $-13 * 26$. Ne koristiti modifikovani Butov algoritam!

-13=11110011	26=00011010				
M	A	P	P ₀	P'	
11110011	00000000	00011010	0	0	M=-13, A=0, P=26, P'=0
11110011	00000000	00011010	0	0	P ₀ P'=00, bez akcije
11110011	00000000	00001101	0	0	APP'→, kraj 1.koraka
11110011	00001101	00001101	0	0	P ₀ P'=10, A=A-M
11110011	00000110	10000110	1	1	APP'→, kraj 2.koraka
11110011	11111001	10000110	1	1	P ₀ P'=01, A=A+M
11110011	11111100	11000011	0	0	APP'→, kraj 3.koraka
11110011	00001001	11000011	0	0	P ₀ P'=10, A=A-M
11110011	00000100	11100001	1	1	APP'→, kraj 4.koraka
11110011	00000100	11100001	1	1	P ₀ P'=11, bez akcije
11110011	00000010	01110000	1	1	APP'→, kraj 5.koraka
11110011	11110101	01110000	1	1	P ₀ P'=01, A=A+M
11110011	11111010	10111000	0	0	APP'→, kraj 6.koraka
11110011	11111010	10111000	0	0	P ₀ P'=00, bez akcije
11110011	11111101	01011100	0	0	APP'→, kraj 7.koraka
11110011	11111101	01011100	0	0	P ₀ P'=00, bez akcije
11110011	<u>11111110</u>	<u>10101110</u>	0	0	APP'→, kraj 8.koraka

Rezultat je u registrima AP i jednak je $(111111010101110)_2 = -338$

3. Izračunati:

a) $-7896+4984$ u BCD kodu 8421.

b) $-7896-4984$ u BCD kodu višak 3.

Brojeve zapisivati sa 5 binarno kodiranih dekadnih cifara.

- a) $-7896+4984 = -(7896-4984)$ jer se oduzimanje vrši nad brojevima zapisanim u obliku znak i apsolutna vrednost

$$X = 7896 \quad Y = 4984$$

Y	0000	0100	1001	1000	0100
$[-Y]_{nk}$	1001	0101	0000	0001	0101
+1					0001
$[-Y]_{pk}$	1001	0101	0000	0001	0110
$S = X + [-Y]_{pk}$					
X	0000	0111	1000	1001	0110
$[-Y]_{pk}$	1001	0101	0000	0001	0110
P'	0	0	0	0	0
S'	1001	1100	1000	1010	1100
P''	1	1	0	1	0
K	0110	0110	0000	0110	0110
S	0000	0010	1001	0001	0010

U skladu sa pravilima za sabiranje brojeva u potpunom komplementu, pojava prenosa p''_5 ne označava prekoračenje.

Dakle: $-7896+4984=-2912$.

b) $-7896-4984=-(7896+4984)$

$X = 7896 \quad Y = 4984$

X	0011	1010	1011	1100	1001
Y	0011	0111	1100	1011	0111
P'	0	1	1	1	0
S'	0111	0010	1000	1000	0000
K	1101	0011	0011	0011	0011
S	0100	0101	1011	1011	0011

Dakle, $-7896-4984=-12880$.

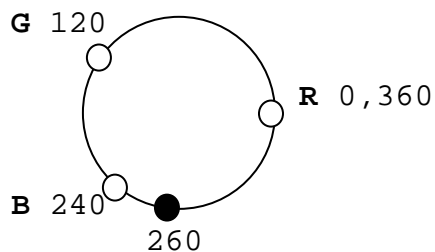
4. a) Prevesti HSB(260, 50%, 40%) u RGB kolor model.
b) Kako izgleda boja HSB(0,0%,100%)?
c) Kako izgleda boja HSB(0,0%,0%)?

a) $H_0 = 60 * (B_2 - B_3) / (B_1 - B_3)$
H = vrednost tona za komp.1 + odklon ka komp.2
 $S = (B_1 - B_3) / B_1$
 $B = B_1 / \text{otklon}$
otklon = 255

H = 260
S = 50%
B = 40%

$B_1 \leq B_2 \leq B_3$ pa otuda sledi da je $0 \leq H_0 \leq 60$

$B_1 = B * \text{otklon} = 0.4 * 255 = 102$
 $B_3 = (1 - S) * B_1 = (1 - 0.5) * 102 = 0.5 * 102 = 51$



Sa slike se vidi da je:

$H = 260 = 240 + 20$

te je odklon $H_0 = 20$, komponenta 1 ima ton 240 (to je B), a komponenta 2 je R (ton 360), pa je preostala komponenta G:

$B_1 = B$
 $B_2 = R$
 $B_3 = G$

$H_0 = 20 = 60 * (B_2 - 51) / (102 - 51)$

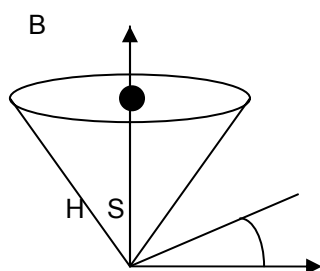
$1 = 3 * (B_2 - 51) / 51$

$1 = (B_2 - 51) / 17$

$17 = B_2 - 51$

$B_2 = 68$

Dakle, RGB = (68, 51, 102).



- b) boja HSB(0,0%,100%) je bela (centar osnovne kupe).
- c) boja HSB(0,0%,0%) je crna (vrh kupe).

5. a) Proveriti da li je niska bitova 101101100111011 ispravno primljena ako je za kodiranje korišćen algoritam *Cyclic Redundancy Check*? Može li se zaključiti koja je originalna niska?
 b) Koja niska bitova će se dobiti nakon kodiranja iste niske 101101100111011 kao u delu a) algoritmom *Cyclic Redundancy Check*?
 U oba slučaja koristiti polinom generator $G(x) = x^3 + x + 1$.

a) Kako je polinom generator $G(x) = x^3 + x + 1$, njegovi koeficijenti čine nisku 1011. Provera da li je niska bitova 101101100111011 primljena ispravno vrši se deljenjem polinoma čiji koeficijenti čine datu nisku polinomom generatorom (po modulu 2).

```

101101100111011
1011
-----
  1100
  1011
  ----
   1111
   1011
   ----
    1001
    1011
    ----
     1010
     1011
     ----
      111
  
```

Kako je ostatak pri deljenju različit od 0, zaključuje se da niska bitova 101101100111011 **nije** ispravno primljena.

Ne može se odrediti koja je originalna niska, jer CRC algoritam ne daje mogućnost lokalizovanja greške, već samo konstatovanja da greška postoji.

Napomena: u slučaju da je niska bitova ispravno primljena (ostatak pri deljenju u gore navedenom obliku je različit od 0), mogla bi se rekonstruisati originalna niska odbacivanjem poslednja tri bita primljene niske (jer je polinom generator stepena $k=3$).

b) Kodiranje niske(poruke) algoritmom CRC vrši se tako što se na originalnu poruku dopiše k koeficijenata ostatka pri deljenju polinoma po sledećoj formuli:

$$(M(x) \cdot x^k) / G(x)$$

$M(x)$ je polinom čiji koeficijenti čine nisku koja treba da se kodira, $G(x)$ je polinom generator, a k stepen polinoma generatora.

Množenje $M(x) \cdot x^k$ ekvivalentno je dopisivanju k nula s desna na nisku koja se kodira, čime se dobija niska 101101100111011000.

Potom se polinom čiji koeficijenti čine novodobijenu nisku deli polinomom generatorom (po modulu 2).

```

101101100111011000
1011
-----
  1100
  1011
  ----
   111
  
```

```

1111
1011
 1001
 1011
  1010
  1011
   1110 ← u ovom koraku se završava postupak deljenja u delu a) bez spuštanja nule
   1011
    1010
    1011
     10

```

Kodiranjem se k koeficijentata ostatka (010) dopisuje s desna na originalnu nisku, tako da je kodirana niska jednaka 101101100111011**010**.

Napomena: primetiti da postupak deljenja u delu b) ima tri koraka više u odnosu na postupak deljenja u delu a), što odgovara dopisanim nulama. Prethodni koraci su identični u oba postupka, tako da se rešenje dela a) može iskoristiti u delu b).

6. Formirati tablicu Hamingovih SEC kodova za 8-bitne reči i izvršiti korekciju greške (ukoliko postoji) za reč

$m_8 \ m_7 \ m_6 \ m_5 \ m_4 \ m_3 \ m_2 \ m_1 \ C_4 \ C_3 \ C_2 \ C_1$
 0 1 0 0 1 0 0 1 1 0 0 0

Iz tabele:

12	1100		M_8
11	1011		M_7
10	1010		M_6
9	1001		M_5
8	1000	C_4	
7	0111		M_4
6	0110		M_3
5	0101		M_2
4	0100	C_3	
3	0011		M_1
2	0010	C_2	
1	0001	C_1	

se dobija da je:

$$C_1 = M_1 \oplus M_2 \oplus M_4 \oplus M_5 \oplus M_7$$

$$C_2 = M_1 \oplus M_3 \oplus M_4 \oplus M_6 \oplus M_7$$

$$C_3 = M_2 \oplus M_3 \oplus M_4 \oplus M_8$$

$$C_4 = M_5 \oplus M_6 \oplus M_7 \oplus M_8$$

gde \oplus označava operaciju ekskluzivne disjunkcije.

Za datu reč dobijaju se sledeće kontrolne cifre:

$$C_1' = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$C_2' = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$C_3' = 0 \oplus 0 \oplus 1 \oplus 0 = 1$$

$$C_4' = 0 \oplus 0 \oplus 1 \oplus 0 = 1$$

$$K' = C_4' C_3' C_2' C_1' = 1111$$

$$K = C_4 C_3 C_2 C_1 = 1000$$

Sindrom reč se dobija poređenjem K i K', tj.

$$C_4' C_3' C_2' C_1' = 1111$$

$$C_4 C_3 C_2 C_1 = \oplus \begin{array}{r} 1000 \\ 0111 \end{array}$$

Odavde se dobija da postoji greška u zapisu koja se nalazi na poziciji 4, tj. na bitu M_4 . Korektna vrednost podatka je: 0100001.

7. Dat je tekst dužine 60 znakova u kome se 3 puta pojavljuje slovo A, 10 puta slovo B, 5 puta slovo C, 18 puta slovo D, 14 puta slovo E i 10 puta slovo F. Odrediti Hafmanove kodove za slova u datom zapisu.

Postupak generisanja Hafmanovog drveta dat je u tabelama.

D	E	F	B	C	A
18	14	10	10	5	3

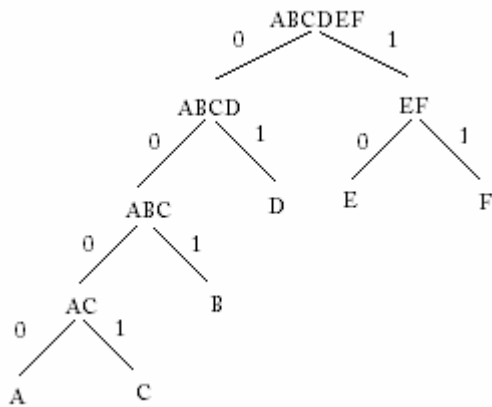
D	E	F	B	AC
18	14	10	10	8

D	ABC	E	F
18	18	14	10

EF	D	ABC
24	18	18

EF	ABCD
24	36

ABCDEF
60



Hafmanovi kodovi:

- A - 0000
- B - 001
- C - 0001
- D - 01
- E - 10
- F - 11

8. a) Navesti razlike između kontinualnih i diskretnih računskih sredstava.
b) Nabrojati događaje iz mehaničkog perioda razvoja informacionih tehnologija.

a) Karakteristike diskretnih računskih sredstava su:

- Svaka cifra broja se registruje u odvojenom objektu kao jedno od njegovih diskretnih stanja. Obično se objekat koji poseduje diskretna stanja naziva ćelija. Diskretna stanja objekta moraju da budu stabilna i moraju međusobno da se razlikuju. Za diskretno stanje se kaže da je stabilno ako se prelazak u drugo diskretno stanje dešava isključivo kao rezultat spoljašnjeg uticaja.
- Tačnost dobijenog rezultata ne zavisi od preciznosti izrade računskog sredstva.
- Diskretna računska sredstva mogu da rešavaju opšte probleme, odnosno mogu da se programiraju.
- Brzina izračunavanja rezultata kod diskretnih računskih sredstava zavisi od složenosti problema koji se rešava.

Karakteristike kontinualnih računskih sredstava su:

- Matematičke veličine se prikazuju sa onom tačnošću koja odgovara mogućnosti preciznog merenja odgovarajuće fizičke veličine.
- Tačnost dobijenog rezultata zavisi od preciznosti izrade računskog sredstva.
- Kontinualna računska sredstva ne mogu da rešavaju opšte probleme, odnosno nisu programabilna.
- Složenost matematičkog modela ne utiče na brzinu dobijanja rezultata

b) Najznačajniji događaji u mehaničkom periodu razvoja informacionih tehnologija su:

- Pojava štamparske presa sa pokretnim slogovima od metala (Johan Gutenberg, 1450. godine)
- Uvođenje logaritama i decimalne tačke u zapisima brojeva (Džon Neper, 1614. godine)
- Pojava klizajućeg lenjira odn. šibera (Viljem Outred, 1622. godine.)
- Konstrukcija mašine za sabiranje i oduzimanje (Šikard, 1623. godine)
- Blez Paskal 1642. godine konstruiše računsku mašinu nazvanu Paskalina
- Pojava mašine koja je mogla da vrši sabiranje, oduzimanje, množenje i deljenje (Lajbnic, 1673. godine)
- Prva upotreba bušene kartice (Žakardov automatski razboj, 1801. godine)
- Mašine Čarlsa Bebidža (diferencijska, nacrt 1822, prototip 1832, nacrt analitičke mašine 1833. godine)

9. Nabrojati karakteristike prve i druge generacije elektronskih računara.

Karakteristike računara I generacije (1939– 1958) su

- logički elementi su se pravili od vakuumskih cevi
- U/I uređjaji su bili bušene kartice, papirne i magnetne trake
- unutrašnju memoriju su činile odložene linije, magnetne trake i magnetni doboši
- za programiranje se koristio mašinski jezik, pri kraju perioda i assembler
- računari su bili veliki, proizvodili su mnogo toplote što je dovođilo do pregoreanja vakuumskih cevi
- u ovom periodu razvijen je FORTRAN - prvi viši programski jezik (1957.g.)
- računari su se koristili u velikim državnim inistitucijama i u vojne svrhe
- preteče operativnih sistema i sistemskog softvera su počele da se javljaju krajem perioda
- krajem perioda je rayvijena i prva globalna računarska mreža (1958. g.)

Karakteristike računara II generacije (1959-1964) su:

- Procesor se pravi od tranzistora
- Unutrašnja memorija je napravljena od magnetnih jezgara
- U/I uređjaji su bušene kartice, papirne i magnetne trake.
- Kao spoljašnja memorija koriste se magnetni diskovi
- Dalji razvoj viših programskih jezika (Lisp, Algol-60, Cobol, ...).
- U ovom periodu dolazi i do početka razvoja operativnih sistema

10. a) Nabrojati načine merenja brzine obrade podataka u računaru.
b) Nabrojati i opisati ulazne uređaje osetljive na dodir.
c) Nabrojati vrste štampača koje poznajete. Kako se formira otisak kod svake od njih?

a) Brzina obrade podataka u računaru se meri prema:

1. broju mašinskih instrukcija koje CPU može da obradi u jednoj sekundi (MIPS),
2. broju operacija u pokretnom zarezu koje mogu da se obrade u jednoj sekundi (FLOPS),
3. vremenu potrebnom za izvršavanje jednog instrukcionog ciklusa,
4. propusnosti, odnosno broju programa koji mogu da se izvrše u određenom vremenskom intervalu,
5. broju izvršenih transakcija u sekundi.

b) Uređaji osetljivi na dodir su:

1. uređaji zasnovani na perima za označavanje, među koje spadaju elektronske table (koje korisniku omogućavaju da putem menija izabere željenu opciju), uređaji koji se koriste kod notebook računara i koji sadrže elektronsko pero i tablet za označavanje (koji korisniku omogućavaju da piše po ekranu) i PDA uređaji,
2. ekrani osetljivi na dodir (koji korisniku omogućavaju da pritiskom na njih bira željene opcije),
3. tableti za digitalizovanje (koji kombinuju elemente uređaja za kontrolu kursora, pera za označavanje i ekrana osetljivih na dodir).

c) Prema tome da li se otisak formira direktnim fizičkim kontaktom mehanizma za štampanje sa papirom ili ne, štampače delimo u dve grupe:

1. bez dodirnog mehanizma i
2. sa dodirnim mehanizmom.

U prvu grupu spadaju:

1. matrični (ili iglični) štampači kod kojih se na pokretnoj glavi štampača u obliku matrice nalaze mali čekići ili tzv. iglice. Otisak se formira pokretanjem iglica koje odgovaraju karakteru koji se štampa.
2. štampači sa lancem i štampači sa trakom koji sadrže karaktere na rotirajućem lancu (odnosno, traci) sa više čekića iza njega. Otisak se formira pozicioniranjem lanca ili trake pritiskom odgovarajućih čekića.

U drugu grupu spadaju:

1. laserski štampači kod kojih laser formira šrafuru na rotirajućem valjku, koji potom dolazi u dodir sa tonerom, te sa papirom.
2. ink-jet štampači koji formiraju otisak raspršivanjem naelektrisanih kapljica boje na papir kroz više otvora.
3. štampači sa toplotnim prenosom otiska koji koriste toplotu za formiranje slike sačinjene od sitnih tačaka na traci kojom se otisak prenosi na papir.

11. a) Navesti i opisati načine pristupa memoriji i navesti primere memorija kod kojih se svaki od njih koristi.
b) Navesti osnovne karakteristike magnetnih diskova. Od kojih parametara zavisi vreme pristupa disku?

a) Mogući načini pristupa unutrašnjoj memoriji su:

1. sekvencijalni pristup (podaci se upisuju u redosledu unošenja, a čitaju u istom ili obratnom redosledu bez preskakanja) – magnetna traka,
2. direktan pristup (na osnovu adrese sloga, pristupa se tačnoj lokaciji ili njenoj okolini) – magnetni diskovi,
3. slučajni pristup (svakoj adresibilnoj lokaciji se na osnovu jedinstvene adrese pristupa direktno bilo da je reč o upisu ili čitanju podataka) – glavna memorija računara,

4. asocijativni pristup (svaka reči se prilikom upisa dodeljuje odgovarajuća maska koja omogućava da joj se prilikom čitanja pristupa na osnovu njenog sadržaja, a ne lokacije) – keš memorija.

b) Osnovne karakteristike magnetnih diskova su:

1. broj upisno-čitajućih glava,
2. mogućnost samo jednostranog ili zapisa podataka sa obe strane diska,
3. fiksnost ili izmenjivost diska.

Vreme pristupa disku zavisi od

- vremena traženja – vreme potrebno za pomeranje ruke sa glavom na cilindar koji sadrži željeni sektor i
- rotacionog kašnjenja – vreme čekanja da se usled rotacije željeni sektor pozicionira ispod glave.

12. Zapisati sledeće brojeve u navedenim osnovama:

a) $(107.65)_{10} = (\dots)_2$

b) $(FC.2E1)_{16} = (\dots)_8$

c) $(2385)_{10} = (\dots)_{16}$

d) $(10110)_2 = (\dots)_3$ bez međuprevođenja u dekadni sistem

a) Prevođe celobrojnog i razlomljenog dela dobijamo deljenjem, odnosno množenjem osnovom ciljnog sistema, pri čemu sve operacije obavljamo u polaznom sistemu:

X_i	107	53	26	13	6	3	1
y_i	1	1	0	1	0	1	1

smer čitanja ←

X_i	0.65	0.30	0.60	0.20	0.40	0.80	0.60
y_i	0	1	0	1	0	0	1

→ smer čitanja

Pri čemu se niska 1001 periodično ponavlja. Dakle, $(107.65)_{10} = (1101011,10100110011001\dots)_2$.

b) Budući da važi $16=2^4$ i $8=2^3$, rešenje dobijamo međuprevođenjem u binarni sistem (prevođenjem svake pojedinačne cifre):

$$(FC.2E1)_{16} = (1111 | 1100.0010 | 1110 | 0001)_2 = (011 | 111 | 100.001 | 011 | 100 | 001)_2 = (374.1341)_8.$$

c) Prevod dobijamo deljenjem osnovom ciljnog sistema, pri čemu sve operacije obavljamo u polaznom sistemu:

X_i	2385	149	9
y_i	1	5	9

smer čitanja ←

Dakle, $(2385)_{10} = (951)_{16}$.

d) Prevod dobijamo deljenjem osnovom ciljnog sistema, pri čemu sve operacije obavljamo u polaznom sistemu. Budući da važi $(3)_{10} = (11)_2$, dobijamo:

X_i	10110	111	10
y_i	1	1	10

smer čitanja ←

Svi rezultati su dobijeni prema pravilima za deljenje neoznačenih celih brojeva:

$$\begin{array}{r} 10110 : 11 = 111 \\ \underline{11} \\ 101 \\ \underline{11} \\ 100 \\ \underline{11} \\ 1 \end{array}$$

$$\begin{array}{r} 111 : 11 = 10 \\ \underline{11} \\ 1 \end{array}$$

$$\frac{10}{10} : 11 = 0$$

Pošto je $(10)_2 = (2)_3$, dobijamo $(10110)_2 = (211)_3$.

13. Kako se vrši sabiranje i oduzimanje brojeva u obliku znak i apsolutna vrednost i potpuni komplement? Kako se vrši kontrola prekoračenja u tim slučajevima?

Znak i apsolutna vrednost:

- Sabiranje:
 1. Ukoliko su brojevi istog znaka taj znak je i znak rezultata. Apsolutna vrednost zbira dobija se sabiranjem apsolutnih vrednosti sabiraka prema pravilima za sabiranje neoznačenih brojeva. Ako se pri sabiranju apsolutnih vrednosti javi prekoračenja tada se prekoračenje javlja i u konačnom zbiru.
 2. Ukoliko su brojevi različitog znaka, znak rezultata je isti kao i znak sabirka koji ima veću apsolutnu vrednost. Apsolutna vrednost zbira dobija se oduzimanjem apsolutne vrednosti manjeg od apsolutne vrednosti većeg sabirka. Pri sabiranju brojeva različitog znaka ne dolazi do prekoračenja.
- Oduzimanje se vrši po pravilima za sabiranje uz promenu znaka drugom operandu uz opisanu kontrolu prekoračenja.

Potpuni komplement:

- Sabiranje: u prvom koraku se brojevi sabiraju prema pravilima za sabiranje neoznačenih brojeva pri čemu se ne vrši kontrola prekoračenja. U drugom koraku se eventualni prenos sa mesta najveće težine briše iz međuzbira čime se dobija konačan rezultat. Prekoračenje se javlja ukoliko se prilikom sabiranja dva broja istog znaka dobija rezultat različitog znaka.
- Oduzimanje se vrši po pravilima za sabiranje uz promenu znaka drugom operandu uz opisanu kontrolu prekoračenja.

14. a) Zapisati broj -345901 u nepakovanom i pakovanom obliku u EBCDIC i ASCII kodu.
b) Odrediti količnik i ostatak brojeva $A=-12345$ i $B=+321$ u kodu 8421.

a)

kod	nepakovani zapis	pakovani zapis
EBCDIC	F3F4F5F9F0D1	0345901D
ASCII	5354555950B1	0345901B

b) U pakovanom obliku ovu operaciju možemo da zapišemo kao 12345D:321C. Količnik je negativan, dok njegove cifre određujemo upoređivanjem delioca sa početnim delom deljenika:

012 < 321 da
0123 < 321 da
01234 < 321 ne

Dalje je:

321*1 < 01234 da
321*2 < 01234 da
321*3 < 01234 da
321*4 < 01234 ne

Odakle dobijamo da je prva cifra količnika 3. Kako je $321*3=963$, dobijeni proizvod oduzimamo od početka deljenika i prelazimo na određivanje naredne cifre istim postupkom. Na taj način dobijamo:

```
012345 : 321 = 38
  963
  ---
 2715
 2568
  ---
  147
```

Dakle, količnik je -38, a ostatak +147.