Programiranje 1

Beleške sa vežbi
Školska 2007/2008 godina

Matematički fakultet, Beograd

Jelena Tomašević

October 9, 2007

Sadržaj

1

1.1 Pozicioni brojni sistemi - konverzije

Pozicioni brojni sistemi su oni u kojima se težina cifre (njen udeo u celokupnoj vrednosti broja) određuje na osnovu njene pozicije u broju (što veća pozicija to je veći i udeo u vrednosti broja). Dekadni brojni sistem je pozicioni, dok rimski brojevi predstavljaju sistem koji nije pozicioni.

Kako su računari zasnovani na binarnoj aritmetici a mi smo navikli da radimo sa dekadnim brojevnim sistemom potrebno je obezbediti prevođenje brojeva iz sistema sa osnovom 10 u sistem sa osnovom 2 i obratno.

Da bi to uradili prvo ćemo posmatrati opštiji problem prevođenja brojeva iz sistema sa proizvoljnom osnovom b u sistem sa osnovom 10.

U bazi sa osnovom 10, na koju smo mi navikli, cifre koje koristimo su $0, 1, \ldots, 9$ odnosno od 0 do $10 - 1$. Znači u proizvoljnoj bazi B koristićemo cifre od 0 do $B - 1$.

Najčešće korišćene baze (sem 10) su stepeni dvojke: $2, 8, 16$. U sledećoj tabeli su prikazani nazivi odgovarajućih brojevnih sistema zajedno sa ciframa koje se u njima koriste. [1]

Naziv	Osnova	Cifre
Dekadni	10	$0, 1, 2, 3, 4, 5, 6, 7, 8, 9$
Binarni	2	$0, 1$
Oktalni	8	$0, 1, 2, 3, 4, 5, 6, 7$
Heksadekadni	16	$0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F$

1.2 Prebacivanje iz sistema sa osnovom b u dekadni brojni sistem

Pozicioni brojni sistemi imaju svojstvo da niz od n cifara u sistemu sa osnovom B

$$\delta_n = d_1 d_2 \ldots d_n$$

predstavlja broj

$$\tilde{\delta}_n = \sum_{i=1}^{n} \tilde{d}_i * B^{n-i} = (\ldots(\tilde{d}_1 * B + \tilde{d}_2) * B + \ldots + \tilde{d}_{n-1}) * B + \tilde{d}_n$$

u dekadnom brojnom sistemu, pri čemu $\tilde{d}_i$ predstavlja numeričku vrednost karaktera d_i (odnosno za ako je $d_i = A$ onda je $\tilde{d}_i = 10$).

[1] pri čemu u heksadekadnom sistemu slova A-F imaju redom vrednosti od $10 - 15$

Na ovaj način su jedinstveno predstavljeni svi brojevi od 0 do $B^n - 1$. Ako prvih i cifara označimo sa $\delta_i = d_1 \dots d_i$ onda važi (za $\delta_0 = 0$)

$$\tilde{\delta}_i = \tilde{\delta}_{i-1} * B + \tilde{d}_i, \qquad 0 \leq \tilde{d}_i < B$$

odatle takođe možemo da zaključimo da je:

$$\tilde{\delta}_{i-1} = \tilde{\delta}_i \text{ div } B, \qquad d_i = \tilde{\delta}_i \bmod B \tag{1.1}$$

Znači, ako pretpostavimo da su cifre broja u sistemu B redom $d_1, \dots, d_n$ onda se njegova vrednost u dekadnom sistemu (označimo je sa x) može izračunati na sledeći način:

1. Neka je $x = 0$ i neka je i indeks tekuće cifre, $i = 1$ na početku.
2. Sada uračunavamo tekuću cifru u vrednost broja: $x = x * B + d_i$, $i = i + 1$
3. Ako je $i > n$ znači da smo uračunali sve cifre broja i da smo dobili vrednost x, inače treba da uračunamo sledeću cifru i vraćamo se na korak 2.

Primer 1 *Prevođenje iz osnova* 2, 16 *i* 8 *u osnovu* 10*:*

$(1101)_2 = 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = (13)_{10}$
$(1101)_{16} = 1 * 16^3 + 1 * 16^2 + 0 * 16^1 + 1 * 16^0 = 4096 + 256 + 1 = (4353)_{10}$
$(F9A)_{16} = F * 16^2 + 9 * 16^1 + A * 16^0 = 15 * 16^2 + 9 * 16^1 + 10 * 16^0 = 3840 + 144 + 10 = (3994)_{10}$

Zadatak 1 *Prebacite sledeće brojeve u dekadni brojni sistem (indeks predstavlja osnovu u kojoj su brojevi zapisani):* $(110111100)_2$, $(77)_8$, $(FFFF)_{16}$

1.3 Prebacivanje iz dekadnog brojnog sistema u sistem sa osnovom b

Inverzni algoritam koji računa niz cifara $d_1 \dots d_n = \delta_n$ koje predstavljaju broj $\tilde{\delta}_n$ u pozicionom sistemu sa osnovom B se dobija primenom jednačina 1.1.

Za dati broj $\tilde{\delta}_i = x$ $(0 \leq x < B^n)$, njegova poslednja cifra u datoj reprezentaciji (sa osnovom B) se dobija kao ostatak pri deljenju broja x sa B. Da bi dobili ostale cifre potrebno je da izračunamo celobrojni količnik pri deljenju x sa B i da na njega primenimo isti algoritam.

1. Krećemo od broja x i u svakom koraku računamo $(n - i)$-tu cifru. Na početku $i = 0$.
2. $c_i = x \bmod B$, $x = x \text{ div } B$, $i = i + 1$.
3. Ako je $x = 0$ dobili smo cifre obrnutim redosledom (odnosno $d_i = c_{n-i}$), inače se vraćamo na korak 2.

Primetimo da ovaj algoritam možemo koristiti i za izdvajanje cifara dekadnog broja (proverite!).

Primer 2 ***Prevođenje iz dekadnog u binarni brojni sistem***

$$(26)_{10} = (?)_2$$

Rešenje:
26 / 2 = 13 i ostatak 0
13 / 2 = 6 i ostatak 1
6 / 2 = 3 i ostatak 0
3 / 2 = 1 i ostatak 1
1 / 2 = 0 i ostatak 1
Dakle, rešenje je broj čije se cifre dobijaju tako što se ostaci dobijeni prethodnim postupkom pročitaju obrnutim redosledom tj. $(11010)_2$

Zadatak 2 *Odredite binarnu reprezentaciju sledećih brojeva:* $(54)_{10}$, $(126)_{10}$, $(332)_{10}$.

Primer 3 ***Prevođenje iz dekadnog u oktalni brojni sistem***

$$(181)_{10} = (?)_8$$

Rešenje:
181 / 8 = 22 i ostatak 5
22 / 8 = 2 i ostatak 6
2 / 8 = 0 i ostatak 2
Dakle, rešenje je broj čije se cifre dobijaju tako što se ostaci dobijeni prethodnim postupkom pročitaju obrnutim redosledom tj. $(265)_8$

Zadatak 3 *Odredite oktalnu prezentaciju sledećih brojeva:* $(67)_{10}$, $(336)_{10}$, $(442)_{10}$

Primer 4 ***Prevođenje iz dekadnog u heksadekadni brojni sistem***

$$(181)_{10} = (?)_{16}$$

Rešenje:
181 / 16 = 11 i ostatak 5
11 / 16 = 0 i ostatak 11(heksadekadna cifra B)
Dakle, rešenje će biti broj cije se cifre dobiju tako što se ostaci dobijeni prethodnim postupkom pročitaju unazad tj. $(B5)_{16}$

Zadatak 4 *Odredite heksadekadnu prezentaciju sledećih brojeva:* $(48)_{10}$, $(1336)_{10}$, $(332)_{10}$.

1.4 Rad sa realnim brojevima

Kada radimo sa realnim brojevima možemo posebno posmatrati ceo deo broja i razlomljeni deo broja. Kako smo do sada radili sa celim brojevima posmatrajmo brojeve oblika $x = 0.d_1d_2\ldots d_n$, za koje je $0 \leq x < 1$.

1.4.1 Prevođenje realnih brojeva iz sistema sa osnovom B u dekadni brojni sistem

Neka je $\delta = 0.d_1d_2\ldots d_n$ razlomljeni deo broja x u sistemu sa osnovom B. Tada će njegova vrednost u dekadnom sistemu biti:

$$\tilde{\delta} = \sum_{i=1}^{n} d_i * B^{-i} = \frac{1}{B}(\tilde{d}_1 + \frac{1}{B}(\tilde{d}_2 + \ldots + \frac{1}{B}\tilde{d}_n)\ldots) \qquad (1.2)$$

Pri čemu je $\tilde{d}_i$ numerička vrednost "cifre" d_i. Odatle dobijamo sledeći algoritam:

1. Neka je $x = 0$ (dekadna vrednost broja), $i = 1$ indeks tekuće cifre koju uračunavamo u vrednost broja i $f = \frac{1}{B}$ tekući koeficijent sa kojim množimo cifru.
2. $x = x + d_i * f$, $i = i + 1$, $f = f * \frac{1}{B}$.
3. Ako je $i > n$ znači da smo uračunali sve cifre i da se u x nalazi dekadna vrednost broja, inače se vraćamo na korak 2.

Primer 5 $(0.1101)_2 = 1 * 2^{-1} + 1 * 2^{-2} + 0 * 2^{-3} + 1 * 2^{-4} = (0.6875)_{10}$

Zadatak 5 *Prebacite sledeće brojeve u dekadni brojni sistem (indeks predstavlja osnovu u kojoj su brojevi zapisani):* $(0.1011)_2$, $(0.77)_8$, $(0.FF)_{16}$

1.4.2 Prevođenje realnih brojeva iz dekadnog brojnog sistema u sistem sa osnovom B

Neka je $\delta_i = 0.d_i \dots d_n$. Na osnovu jednačine 1.2 vidimo da važi:

$$\tilde{\delta}_i = \frac{1}{B}(\tilde{d}_i + \tilde{\delta}_{i+1}), \qquad 0 \leq \tilde{\delta}_i < 1$$

Odnosno

$$\tilde{d}_i = \mathrm{trunc}(B * \tilde{d}_i)$$

$$\tilde{\delta}_{i+1} = B * \tilde{\delta}_i - \tilde{d}$$

pri čemu je $\mathrm{trunc}(x)$ ceo deo broja x. Odatle direktno vidimo na koji način možemo izračunati cifre broja u sistemu sa osnovom B i dobijamo sledeći algoritam [2]:

1. Neka je x broj čiji zapis određujemo, $i = 1$ indeks tekuće cifre koju računamo.
2. $d_i = \mathrm{trunc}(B * x)$.
3. $x = B * x - d_i$, $i = i + 1$.
4. Ako je $i = n$ dobili smo n cifara razlomljenog dela broja, inače se vraćamo na korak 2.

Ovim algoritmom se cifre dobijaju u željenom redosledu, odnosno od prve ka poslednjoj.

Primer 6 *Odrediti binarni zapis broja $x = (0.867)_{10}$ na 4 decimale.*

$0.867 * 2 = 1.734$, *ceo deo 1*
$0.734 * 2 = 1.468$, *ceo deo 1*
$0.468 * 2 = 0.936$, *ceo deo 0*
$0.936 * 2 = 1.872$, *ceo deo 1*

Dakle rešenje se dobija tako što se cifre čitaju onim redosledom kojim su dobijene tj. $(0.1101)_2$

1.5 Direktno prevođenje iz binarnog u heksadekadni sistem

Za kodiranje heksadekadnih cifara dovoljne su binarne reči dužine četiri ($16 = 2^4$).

Heksadekadna cifra	Binarni kod	Heksadekadna cifra	Binarni kod
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

[2] pri čemu primetimo da iz petlje izlazimo kada dobijemo željeni broj cifara a ne kada x postane 0 iz razloga što radimo sa realnim brojevima koji ne moraju imati konačan zapis

Primetimo da je na ovaj način svakoj heksadekadnoj cifri jedinstveno dodeljen kod dužine četiri u binarnom sistemu što nam omogućava da obavljamo direktno prevođenje iz binarnog u heksadekadni sistem na sledeći način:

Binarne cifre se grupišu u grupe od 4 cifre, pocev od bitova najmanje težine. Ako ukupan broj bitova nije deljiv sa četiri, onda se dopisuje potreban broj vodećih nula (one su bez uticaja na promenu vrednosti originalnog zapisa).

Primer 7 $(1111011100001101010000)_2 = (\ 0011\ 1101\ 1100\ 0011\ 0101\ 0000)_2 = (3DC350)_{16}$

Zadatak 6 *Odredite heksadekadni zapis sledećeg binarnog broja* $(11010100100)_2$

1.6 Direktno prevođenje iz binarnog u oktalni sistem

Za kodiranje oktalnih cifara dovoljne su binarne reci dužine tri ($8 = 2^3$).

Oktalna cifra	Binarni kod	Oktalna cifra	Binarni kod
0	000	4	100
1	001	5	101
2	010	6	110
3	011	7	111

Sada smo svakoj oktalnoj cifri jedinstveno dodelili binarni kod dužine tri što nam omogućava direktno prevođenje. Binarne cifre se grupišu grupe od po 3 cifre, počev od bitova najmanje težine. Ako ukupan broj bitova nije deljiv sa tri, onda se dopisuje potreban broj vodećih nula.

Primer 8 $(11111010001010)_2 = (011\ 111\ 010\ 001\ 010)_2 = (37212)_8$

Zadatak 7 *Odredite oktalni zapis sledećeg binarnog broja* $(11010100100)_2$