Pretvaranje brojeva iz jednog brojevnog sustava u drugi

Kao što je poznato kod računala se za predstavljanje bilo koje vrste podataka koriste samo dva diskretna nivoa fizikalnih veličina, pa iz toga razloga ljudi za predstavljanje podataka u računalu koriste ***binarne znamenke*** **0** ili **1.**

Brojevni sustav označava *način na koji zapisujemo brojeve pomoću određenog skupa znamenki*. ***Baza brojevnog sustava*** određuje broj simbola (znamenki) koje u zapisu koristimo. Primjerice, ***binarni*** brojevni sustav (*sustav s bazom 2*) ima dvije znamenke: **0** i **1**. ***Oktalni*** brojevni sustav (*sustav s bazom 8*) ima 8 znamenaka: **0**, **1**, **2**, **3**, **4**, **5**, **6** i **7**. ***Heksadecimalni*** brojevni sustav (*sustav s bazom 16*) ima 16 znamenaka: za prvih se deset znamenaka koriste brojevi **0**, **1**, **2**, **3**, **4**, **5**, **6**, **7**, **8** i **9**, dok se za preostalih šest koriste slova engleskog alfabeta **A**, **B**, **C**, **D**, **E** i **F**.[1]

Isti broj zapisan u *različitim brojevnim sustavima* ima *različiti zapis, primjerice, broj* $10_{(10)}$ *u binarnom je brojevnom sustavu* $1010_{(2)}$*, u oktalnom* $12_{(8)}$ *dok je u heksadecimalnom* $A_{(16)}$.

Prevođenje broja iz jednog brojevnog sustava u zapis drugog sustava, obavlja se preko dekadskog brojevnog sustava. Najprije se broj iz zadanog sustava prevede u odgovarajući zapis u dekadskom sustavu, te se potom dobiveni broj iz dekadskog sustava prevodi u odredišnu bazu na način da se sukcesivno dijeli s bazom odredišnog sustava, sve dok konačni međurezultat dijeljenja ne postane jednak nuli. Zapis broja u odredišnom sustavu dobije se potom na način da se ostaci dijeljenja zapišu redoslijedom obrnutim od svoga nastajanja.

U posebnim slučajevima, za pretvaranje brojeva iz jednog u drugi sustav, postoje i ***jednostavnije metode***. Naime, ako se ***baza jednog*** *sustava* može izraziti kao ***potencija druge baze***, tada se transformacija smije obavljati ***znamenku po znamenku***.

Pretvaranje "***po znamenkama***" možemo koristiti za pretvaranje između ***oktalnog*** i ***binarnog*** sustava jer se baza oktalnog sustava može prikazati u obliku potencije broja 2 ($8 = 2^3$), te također, između ***heksadecimalnog*** i ***binarnog****, jer se i baza heksadecimalnog sustava može prikazati u obliku potencije broja 2 (*$16 = 2^4$*)*.

Pretvaranje iz ***oktalnog*** brojevnog sustava u ***binarni*** brojevni sustav

Pretvaranje broja zapisanog u oktalnom brojevnom sustavu u broj zapisan u binarnom sustavu, provodi se na način da se svakoj oktalnoj znamenki u zapisu broja, pridruže po 3 binarne znamenke prema tablici.

[1] U svim sustavima s bazom većom od 10 prvih se deset znamenki označava brojevima od 0 do 9, dok se preostale znamenke prikazuju slovima engl. alfabeta.

Tablica 1. Tablica za pretvaranje brojeva između oktalnog i binarnog brojevnog sustava

Znamenke u oktalnom sustavu (bazi 8)	Binarni zapis		
	4	2	1
	$\mathbf{2^2}$	$\mathbf{2^1}$	$\mathbf{2^0}$
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Primijetimo da se oktalna znamenka 0 binarno zapisuje s 000, jer je to $0x2^0 + 0x2^1 + 0x2^2 = 0x1 + 0x2 + 0x4$, …, oktalna znamenka 6 zapisuje se binarno 110, jer je to $0x2^0 + 1x2^1 + 1x2^2 = 0x1 + 1x2 + 1x4$, a znamenka 7 s 111, jer je to $1x2^0 + 1x2^1 + 1x2^2 = 1x1 + 1x2 + 1x4$.

Također, primijetimo zakonitost u "nizanju" binarnih znamenki: prvi stupac ispod 2^2 ispunjen je s četiri nule i četiri jedinice, drugi stupac ispod 2^1 s dvije nule, dvije jedinice, dvije nule i dvije jedinice, dok je posljednji stupac ispod 2^0 ispunjen naizmjenično s nula i jedan.

Primjer 1.

Broj $4017_{(8)}$ zadan u oktalnom sustavu, pretvorite u broj zapisan u binarnom sustavu.

$$4017_{(8)} = 100\ 000\ 001\ 111_{(2)}$$

Primjer 2.

Broj $105_{(8)}$ zadan u oktalnom sustavu, pretvorite u broj zapisan u binarnom sustavu.

$$105_{(8)} = 001\ 000\ 101_{(2)} = 1\ 000\ 101_{(2)}$$

Dvije vodeće nule slobodno možemo izbrisati (iako nam ne smetaju).

Pretvaranje iz ***binarnog*** brojevnog sustava u ***oktalni*** brojevni sustav

Postupak pretvaranja broja zapisanog u binarnom brojevnom sustavu u broj zapisan u oktalnom brojevnom sustavu sastoji se iz sljedećih koraka:

1. *S desna prema lijevo, grupiraju se po* ***tri*** *binarne znamenke.*
2. *Svakoj skupini od* ***tri*** *binarne znamenke pridruži se po jedna oktalna prema tablici.*

Primjer 3.

Pretvoriti broj **10 100 111 001 (2)** zadan u binarnom brojevnom sustavu, u broj zapisan u oktalnom brojevnom sustavu.

$$\boxed{0}10 \mid 100 \mid 111 \mid 001\ _{(2)} = \ _{(8)}$$
$$2 \quad 4 \quad 7 \quad 1$$

prema potrebi dopišite potreban broj vodećih nula kako biste lakše pročitali znamenku u tablici

$$10\ 100\ 111\ 001_{(2)} = 2471_{(8)}$$

Primjer 4.

Pretvoriti broj **1 111 001 $_{(2)}$** zadan u binarnom brojevnom sustavu u broj zadan u sustavu s bazom 8.

$$\boxed{00}1 \;|\; 111 \;|\; 001 \;_{(2)} = \;_{(8)}$$
$$1 \quad 7 \quad 1$$

prema potrebi dopišite potreban broj vodećih nula kako biste lakše pročitali znamenku u tablici

1 111 001 $_{(2)}$ = 171 $_{(8)}$

Pretvaranje iz ***heksadecimalnog*** brojevnog sustava u ***binarni*** brojevni sustav

Pretvaranje broja zapisanog u heksadecimalnom brojevnom sustavu u broj zapisan u binarnom sustavu, provodi se na način da se svakoj heksadecimalnoj znamenki u zapisu broja, pridruže po 4 binarne znamenke prema tablici.

Tablica 2. Tablica za pretvaranje brojeva između heksadecimalnog i binarnog brojevnog sustava

Znamenke u heksadecimalnom sustavu (u bazi 16)	Binarni zapis			
	8	4	2	1
	2^3	**2^2**	**2^1**	**2^0**
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
A (10)	1	0	1	0
B (11)	1	0	1	1
C (12)	1	1	0	0
D (13)	1	1	0	1
E (14)	1	1	1	0
F (15)	1	1	1	1

Primijetimo da se heksadecimalna znamenka 0 binarno zapisuje s 0000, jer je to $0x2^0 + 0x2^1 + 0x2^2 + 0x2^3 = 0x1 + 0x2 + 0x4 + 0x8$, …, heksadecimalna znamenka E (14) zapisuje se binarno 1110, jer je to $0x2^0 + 1x2^1 + 1x2^2 + 1x2^3 = 0x1 + 1x2 + 1x4 + 1x8$, a znamenka F (15) s 1111, jer je to $1x2^0 + 1x2^1 + 1x2^2 + 1x2^3 = 1x1 + 1x2 + 1x4 + 1x8$.

Također, primijetimo zakonitost u "nizanju" binarnih znamenki prvi stupac ispod 2^3 ispunjen je s osam nula i osam jedinica, drugi ispod 2^2 ispunjen je s četiri nule i četiri jedinice naizmjenično, treći stupac ispod 2^1 s dvije nule, dvije jedinice, naizmjenično, dok je posljednji stupac ispod 2^0 ispunjen naizmjenično s nula i jedan.

Primjer 5.
Broj FACE $_{(16)}$ zadan u heksadecimalnom brojevnom sustavu, pretvorite u broj zapisan u binarnom sustavu.

FACE $_{(16)}$ = 1111 1010 1100 1110 $_{(2)}$

Pretvaranje iz ***binarnog*** brojevnog sustava u ***heksadecimalni*** brojevni sustav

Postupak pretvaranja broja zapisanog u binarnom brojevnom sustavu u broj zapisan u heksadecimalnom brojevnom sustavu sastoji se iz sljedećih koraka:

1. *S desna prema lijevo grupiraju se po* ***četiri*** *binarne znamenke.*
2. *Svakoj skupini od* ***četiri*** *binarne znamenke pridruži se po jedna heksadecimalna znamenka prema tablici.*

Primjer 6.
Broj 1 1100 1011 0110 $_{(2)}$ zadan u binarnom brojevnom sustavu pretvorite u broj zapisan u heksadecimalnom brojevnom sustavu.

1 1100 1011 0110 $_{(2)}$ = 1CB6 $_{(16)}$

Pretvaranje između ***oktalnog*** i ***heksadecimalnog*** brojevnog sustava

Pretvaranje brojeva između ***oktalnog*** i ***heksadecimalnog*** brojevnog sustava obavlja se preko ***binarnog***.

Primjer 7.

Broj 2471$_{(8)}$ pretvorite u broj zapisan u bazi 16.

2471$_{(8)}$ = 010 100 111 001 $_{(2)}$ = 0101 | 0011 | 1001$_{\mathbf{(2)}}$ **=** 539$_{(16)}$

Primjer 8.

Broj 79A1B4F$_{(16)}$ pretvorite u broj zapisan u bazi 8.

79A1B4F $_{(16)}$ = 0111 1001 1010 0001 1011 0100 1111 $_{(2)}$ =
=000 | 111 | 100 | 110 | 100 | 001 | 101 | 101 | 00 1 | 111 $_{(2)}$ =
= 0 7 4 6 4 1 5 5 1 7 $_{(8)}$
= 746 415 517 $_{(8)}$

Zadaci

137272 $_{(8)}$ = BEBA $_{(16)}$

135272 $_{(8)}$ = BABA $_{(16)}$

157332 $_{(8)}$ = DEDA $_{(16)}$

135312 $_{(8)}$ = BACA $_{(16)}$

FACA $_{(16)}$ = 175312 $_{(8)}$

ABBA $_{(16)}$ = 125672 $_{(8)}$

LOGIČKI ISKAZI

Sa stajališta matematičke logike, svaka tvrdnja može poprimiti jednu od dvije moguće vrijednosti: logičku ***istinu*** (engl. TRUE) ili logičku ***neistinu*** - laž (engl. FALSE).

U daljnjem tekstu za označavanje ***logičke istine*** koristit će se ***broj 1***, a za označavanje ***logičke neistine*** koristit će se ***broj 0***.

Logički se iskazi rješavaju s lijeva u desno uvažavajući prioritet logičkih operatora.

Prioritet logičkih operatora:

- $\neg$ (NOT, logička negacija),
- $\wedge$ (AND, konjunkcija),
- $\vee$ (OR, disjunkcija).

Prioritet izvođenja logičkih operacija *mijenja* se uporabom "okruglih"zagrada (), na način da se najprije rješava ono što se unutar njih nalazi.

Tablica 3. Tablica istine operatora negacije

A	**$\neg$A**
0	1
1	0

Tablica 4. Tablica istine operatora konjunkcije[2]

A	**B**	**A $\wedge$ B**
0	0	0
0	1	0
1	0	0
1	1	1

Tablica 5. Tablica istine operatora disjunkcije[3]

A	**B**	**A $\vee$ B**
0	0	0
0	1	1
1	0	1
1	1	1

Napomena.

Primijetimo analogiju između **logičkih** i **aritmetičkih** operacija.

[2] Konjunkcija - AND – daje istinu samo kada su obje tvrdnje istinite (i A i B), pogledajte tablicu istine.

[3] Disjunkcija – OR – daje istinu kada je barem jedna tvrdnja istinita (ili A, ili B ili obje), pogledajte tablicu istine.

Zamijenimo li $\wedge$ s operacijom *množenja*, a $\vee$ s operacijom *zbrajanja*, prioriteti ostaju jednaki, a čak se i rezultati operacija *gotovo* podudaraju (jedina je razlika 1 + 1 = 2 što *nije moguća vrijednost* kod logičkih iskaza, jer je $1 \vee 1 = 1$)

Ovim trikom se uvijek možete poslužiti ako će vam olakšati rješavanje zadataka.[4]

Pri rješavanju tablica istine za zadanu tvrdnju potrebno je uočiti broj varijabli, poredati ih ***abecednim redoslijedom*** i nanizati *sve moguće vrijednosti*. Broj različitih slučajeva u tablici ovisit će o broju varijabli i iznositi će $2^{\text{broj varijabli}}$.
Primijetimo da smo u tablici istine za *negaciju* s obzirom da se radi o unarnom operatoru i jednoj varijabli imali 2^1 slučaja, odnosno samo dva reda u tablici, kod konjunkcije i disjunkcije s obzirom da se radi o binarnim operatorima imali smo 2^2 slučajeva odnosno 4 reda u tablici.

Primjer 1. Napraviti tablicu istine za zadanu tvrdnju:

$$C \vee A \wedge B$$

U zadanoj su tvrdnji tri varijable te će biti 8 različitih slučajeva (2^3). Vrijednosti za A, B i C nanizat ćemo tako da za A najprije dodijelimo 4 nule, potom 4 jedinice, za B po 2 nule i dvije jedinice i za C naizmjenično nula i jedan do kraja. Primijetimo da vrijednosti za A, B i C u tablici zapravo odgovaraju binarnom zapisu oktalnih znamenki od 0 do 7.

A	**B**	**C**	**A ∧ B**	**A ∧ B ∨ C**
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1
1	1	0	1	1
1	1	1	1	1

Primjer 2. Napraviti tablicu istine za zadanu tvrdnju:

$$C \wedge \neg A \wedge D \vee B$$

U zadanoj su tvrdnji četiri varijable te će biti 16 različitih slučajeva (2^4). Vrijednosti za A, B, C i D nanizat ćemo tako da za A najprije dodijelimo 8 nula, potom 8 jedinica, za B po 4 nule, 4 jedinice, 4 nule i 4 jedinice, za C po 2 nule i 2 jedinice do kraja i za D naizmjenično 0 i 1 do kraja. Primijetimo da vrijednosti za A, B, C i D u tablici zapravo odgovaraju binarnom zapisu heksadecimalnih znamenki od 0 do F (15).

Prilikom rješavanja tablice istine, pojedine izraze možemo zamijeniti "novom oznakom" kako ne bismo morali prepisivati cijele izraze, primjerice **C ∧ ¬ A** zamijenili smo s **X**, a **C ∧ ¬ A ∧ D,** odnosno **X** ∧ **D** s **Y**.

[4] Maksimović Srđan, funkcije_sustavi.pdf, str.2

C ∧ ¬ A ∧ D ∨ B

A	**B**	**C**	**D**	**¬A**	$\overbrace{\mathbf{C\wedge\neg A}}^{X}$	$\overbrace{\mathbf{X \wedge D}}^{Y}$	**Y ∨ B**
0	0	0	0	1	0	0	0
0	0	0	1	1	0	0	0
0	0	1	0	1	1	0	0
0	0	1	1	1	1	1	1
0	1	0	0	1	0	0	1
0	1	0	1	1	0	0	1
0	1	1	0	1	1	0	1
0	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	0	0	0	0	1
1	1	0	1	0	0	0	1
1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	1

Primjer 3. Napraviti tablicu istine za zadanu tvrdnju:

A ∨ ¬ D ∧ C ∨ B

A	**B**	**C**	**D**	**¬ D**	**¬ D ∧ C**	**A ∨ ¬ D ∧ C**	**A ∨ ¬ D ∧ C ∨ B**
0	0	0	0	1	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	1	1	1	1
0	0	1	1	0	0	0	0
0	1	0	0	1	0	0	1
0	1	0	1	0	0	0	1
0	1	1	0	1	1	1	1
0	1	1	1	0	0	0	1
1	0	0	0	1	0	1	1
1	0	0	1	0	0	1	1
1	0	1	0	1	1	1	1
1	0	1	1	0	0	1	1
1	1	0	0	1	0	1	1
1	1	0	1	0	0	1	1
1	1	1	0	1	1	1	1
1	1	1	1	0	0	1	1

Primjer 4. Napraviti tablicu istine za zadanu tvrdnju:

D ∨ ¬ B ∧ A ∨ C

A	B	C	D	¬ B	¬ B ∧ A	D ∨ ¬ B ∧ A	D ∨ ¬ B ∧ A ∨ C
0	0	0	0	1	0	0	0
0	0	0	1	1	0	1	1
0	0	1	0	1	0	0	1
0	0	1	1	1	0	1	1
0	1	0	0	0	0	0	0
0	1	0	1	0	0	1	1
0	1	1	0	0	0	0	1
0	1	1	1	0	0	1	1
1	0	0	0	1	1	1	1
1	0	0	1	1	1	1	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	0	1
1	1	1	1	0	0	1	1

Primjer 5. Napraviti tablicu istine za zadanu tvrdnju:

$$D \vee \neg B \wedge A \vee C \wedge B$$

A	B	C	D	¬ B	¬ B ∧ A	D ∨ ¬ B ∧ A	C ∧ B	D ∨ ¬ B ∧ A ∨ C ∧ B
0	0	0	0	1	0	0	0	0
0	0	0	1	1	0	1	0	1
0	0	1	0	1	0	0	0	0
0	0	1	1	1	0	1	0	1
0	1	0	0	0	0	0	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	0	1	1
0	1	1	1	0	0	1	1	1
1	0	0	0	1	1	1	0	1
1	0	0	1	1	1	1	0	1
1	0	1	0	1	1	1	0	1
1	0	1	1	1	1	1	0	1
1	1	0	0	0	0	0	0	0
1	1	0	1	0	0	1	0	1
1	1	1	0	0	0	0	1	1
1	1	1	1	0	0	1	1	1

Primjer 6. Napraviti tablicu istine za zadanu tvrdnju:

$$\neg (\neg A \wedge D \vee C \wedge B)$$

A	**B**	**C**	**D**	**¬ A**	**¬ A ∧ D**	**C ∧ B**	**¬ A ∧ D ∨ C ∧ B**	**¬ (¬ A ∧ D ∨ C ∧ B)**
0	0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	1	0
0	0	1	0	1	0	0	0	1
0	0	1	1	1	1	0	1	0
0	1	0	0	1	0	0	0	1
0	1	0	1	1	1	0	1	0
0	1	1	0	1	0	1	1	0
0	1	1	1	1	1	1	1	0
1	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	1
1	0	1	0	0	0	0	0	1
1	0	1	1	0	0	0	0	1
1	1	0	0	0	0	0	0	1
1	1	0	1	0	0	0	0	1
1	1	1	0	0	0	1	1	0
1	1	1	1	0	0	1	1	0