Introducere

În zilele noastre calculatorul a devenit nelipsit în majoritatea activităţilor cotidiene. Pentru un bun informatician este necesar nu numai să-l folosească, dar şi să-i înţeleagă modul de funcţionare. Cursul de *Logică computaţională* este o prezentare a fundamentelor aritmetice şi logice ale sistemelor de calcul, care, în general, rămân neschimbate, deşi tehnologiile folosite la fabricarea calculatoarelor sunt într-o continuă schimbare.

Obiectivele cursului

Cursul de *Logică computaţională* are ca obiectiv principal îmbogăţirea cunoştinţelor de fundamentele Informaticii ale studenţilor Programului de studii Informatică, forma de învăţământ ID. În acest sens, la sfârşitul acestui curs, studenţii vor fi capabili să:

- Efectueze operaţii aritmetice cu numere reprezentate în orice bază;
- Să efectueze conversii între diferite baze;
- Să identifice legăturile dintre calculatoare şi sistemele de numeraţie binare, octale şi hexazecimale;
- Să ştie cum sunt reprezentate în memoria calculatorului numerele întregi, reale şi caracterele;
- Să interpreteze conţinutul unei locaţii de memorie, cunoscând tipul de date memorate;
- Să determine forma normală disjunctivă şi forma normală conjunctivă a unei funcţii Booleene;
- Să minimize funcţii Boolene;
- Să construiască circuitul logic combinatorial optim pentru orice problemă practică care poate fi scrisă printr-o funcţie Booleană.

Cerinţe preliminare

Nu sunt necesare decât cunoştinţe din învăţământul preuniversitar, acesta fiind unul dintre cursurile din primul semestru.

Resurse

Parcurgerea unităţilor de învăţare aferente acestui curs nu necesită existenţa unor mijloace sau instrumente de lucru.

Structura cursului

Cursul de *Logică computaţională* este structurat în patru module, astfel: primul modul cuprinde trei unităţi de învăţare, al doilea modul trei unităţi de învăţare, al treilea modul cinci unităţi de învăţare, iar al patrulea modul cunprinde două unităţi de învăţare. La rândul său, fiecare unitate de învăţare cuprinde: obiective, aspecte teoretice privind tematica unităţii de învăţare respective, exemple, teste de evaluare precum şi probleme propuse spre discuţie şi rezolvare.

La sfârşitul ultimelor trei module sunt indicate teme de control. Rezolvarea acestor trei teme de control este obligatorie. Acestea vor fi transmise de către studenţi profesorului până la odată prestabilită.

Durata medie de studiu individual

Parcurgerea de către studenţi a unităţilor de învăţare ale cursului de *Logică computaţională* (atât aspectele teoretice cât şi rezolvarea testelor de autoevaluare şi rezolvarea problemelor propuse) se poate face în 2-3 ore pentru fiecare unitate de învăţare.

Evaluarea

La sfârşitul semestrului, fiecare student va primi o notă, care va cuprinde: un test, ce va conţine întrebări teoretice şi practice din materia prezentată în cadrul acestui material, test de va deţine o pondere de 75% în nota finală. Media notelor aferente testelor de evaluare, realizate pe parcursul semestrului, va deţine o pondere de 25%.

Spor la treaba !

Cuprins

Modulul 1. Sisteme de numeraţie

Cuprins

Introducere

Sistemele de numeraţie au apărut pentru a răspunde necesităţii oamenilor de a număra. Dacă iniţial oamenii peşterilor făceau doar diferenţa dintre *unu* şi *mulţi*, cu timpul limbile primitive au evoluat astfel încât se putea face distincţia între *unu*, *doi* şi *mulţi*, iar mai târziu între *unu*, *doi*, *trei* şi *mulţi*. Adică nu existau cuvinte care să exprime numere mai mari decât trei. Acestă lipsă mai persistă şi astăzi în unele limbi, de exemplu în limba vorbită de indienii Siriona din Bolivia.

Deşi nu aveau cuvinte care să exprime numere mai mari decât 2, anumite triburi (Bacairi şi Bororo din Brazilia) au creat sisteme numerice de genul *unu*, *doi*, *doi şi unu*, *doi şi doi*, *doi si doi şi unu* şi aşa mai departe. Acest sistem de numeraţie nu este altceva decât binecunoscutul sistem binar atât de răspândit în era calculatoarelor. Însă puţini oameni au numărat din 2 în 2. O altă modalitate de numărare era cu ajutorul unor crestături făcute într-un os. Crestăturile mici erau grupate câte 5, iar după 5 grupe exista o crestătură de alt tip. Acestă modalitate de grupare a crestăturilor în os corespunde sistemului de numeraţie în baza 5, care a fost preferat de multe culturi, fiind mai uşor de utilizat deoarece, pentru a număra, oamenii îşi foloseau degetele, care sunt grupate tot câte 5. Folosind apoi degetele ambelor mâini s-a ajuns la sistemul zecimal, iar folosindu-le şi pe cele ale picioarelor la cel în baza 20, utilizat de mayaşi. De asemenea, locuitorii Franţei se pare că au folosit sistemul în baza 20.

Un sistem care merită menţionat este cel babilonian care era sexagesimal, adică în baza 60, care, în mod bizar, nu avea 60 de simboluri diferit pentru fiecare „cifră” . Babilonienii nu foloseau decât două semne: un cui pentru 1 şi două cuie pentru 10. Celelalte „cifre” erau grupe de astfel de semne. Dar o mare problema a sistemului babilonian a fost faptul că un cui putea reprezenta 1, 60, 3600 sau o infinitate de alte numere.

În zilele noastre, este util să putem reprezenta numere şi să putem lucra cu numere nu numai în baza 10, dar şi în orice alt sistem de numeraţie şi să putem trece dintr-o bază în alta.

Competenţe

La sfârşitul acestui modul studenţii vor fi capabili să:

- Reprezinte orice număr într-o bază dată *b*;
- Efectueze schimbări de baze;
- Efectueze operaţii aritmetice cu numere reprezentate în orice bază;
- Să identifice legăturile dintre calculatoare şi sistemele de numeraţie binare, octale şi hexazecimale.

Unitatea de învăţare M1.U1. Conceptul de sistem de numeraţie

Cuprins

M1.U1.1. Introducere

Sistemele de numeraţie au apărut pentru a răspunde necesităţii oamenilor de a număra. În decursul timpului au fost folosite diferite sisteme de numeraţie, până ca sistemul zecimal să se impună.

M1.U1.2. Obiectivele unităţii de învăţare

Această unitate de învăţare îşi propune ca obiectiv principal însuşirea de către studenţi a conceptului de sistem de numeraţie.

La sfârşitul acestei unităţi de învăţare studenţii vor fi capabili să:

- realizeze deosebirile dintre sistemele de numeraţie poziţionale şi cele nepoziţionale;
- Reprezinte orice număr într-o bază dată *b*.

Durata medie de parcurgere a primei unităţi de învăţare este de 3 ore.

M1.U1.3. Definirea sistemului de numeraţie

Un *sistem de numeraţie* este format din totalitatea regulilor de reprezentare a numerelor cu ajutorul unor simboluri distincte, numite *cifre*. Există două categorii de sisteme de numeraţie:

1. sisteme de numeraţie poziţionale
2. sisteme de numeraţie aditive sau nepoziţionale

Un sistem de numeraţie se numeşte *poziţional*, dacă valoarea unei cifre este dată de poziţia pe care aceasta o ocupă în cadrul numărului. Sistemele de numeraţie zecimal, binar, octal, hexazecimal sunt sisteme de numeraţie poziţionale.

Într-un sistem de numeraţie *aditiv*, cifrele au aceeaşi valoare, indiferent de poziţia pe care se află. De exemplu, sistemul de numeraţie roman este un sistem de numeraţie aditiv.

Numărul de cifre dintr-un sistem de numeraţie poziţional se numeşte *bază*.

Exemple de sisteme de numeraţie poziţionale

1. Sistemul binar are baza 2, iar cifrele sale sunt 0 si 1, numite şi *biţi* (bit = binary digit). Sistemul binar este utilizat pentru reprezentarea cele două stări posibile ale unui dispozitiv bistabil. De exemplu, un întrerupător poate fi deschis sau închis, un loc de pe un disc magnetic pot fi magnetizat sau nemagnetizat etc.
2. Sistemul octal are baza 8, iar cifre sale sunt 0, 1, 2, 3, 4, 5, 6 şi 7. A fost folosit şi este folosit în continuare în limbajul de asamblare.
3. Sistemul zecimal este sistemul de numeraţie pe care îl folosim în viaţa de zi cu zi, are la baza 10, iar cifre sale sunt 0, 1, 2, 3, 4, 5, 6, 7, 8 şi 9.
4. Sistemul hexazecimal are baza 16 şi cifre sale sunt 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E şi F. Sistemul de numeraţie hexazecimal este utilizat atunci când se lucrează direct cu adresele de memorie.

Exemplu de sistem de numeraţie aditiv

Sistemul roman este un sistem de numeraţie aditiv. Cifrele sale sunt I, V, X, L, C, D, M, ale căror valori sunt 1, 5, 10, 50, 100, 500, 1000.

Reguli de scriere:

1. mai multe cifre de aceeaşi valoare, scrise consecutiv, reprezintă suma acestor cifre. De exemplu, II = 2, XXX = 30.
2. două cifre diferite, cu cifra mai mare aflată în faţa cifrei mai mici, reprezintă suma acestor cifre. De exemplu, XI = 11, CX = 110.
3. două cifre diferite, cu cifra mai mare aflată după cifra mai mică, reprezintă diferenţa acestor cifre. De exemplu, IX = 9, XC = 90.

Sistemul roman are câteva dezavantaje majore: sunt posibile mai multe reprezentări ale aceluiaşi număr (VIII = IIX), reprezentări lungi ale unor numere relativ “mici” (CCCXXXIII = 333), operaţii aritmetice foarte dificil de efectuat, lipsa unei reprezentări pentru 0 etc.

M1.U1.4. Teorema sistemelor de numeraţie

Teorema sistemelor de numeraţie Fie $b \in \mathbf{N}^*$, $b > 1$. Pentru orice număr natural $N \in \mathbf{N}^*$, există numerele $n, a_0, a_1, ..., a_n \in \mathbf{N}$ astfel încât $a_i < b$ pentru orice i, $0 \le i \le n$, $a_n > 0$ şi

$$N = a_n b^n + a_{n-1} b^{n-1} + ... + a_1 b + a_0.$$

Mai mult, aceste numere sunt unice.

Demonstraţie. Din Teorema împărţirii cu rest avem că:

$$N = N_0 b + a_0 \text{ cu } 0 \le a_0 < b.$$

Evident, $N_0 < N$, pentru că în caz contrar am avea

$$N < Nb \le N_0 b \le N_0 b + a_0 = N, \text{ ceea ce este absurd.}$$

De asemenea, din Teorema împărţirii cu rest avem că:

$$N_0 = N_1 b + a_1 \text{ cu } 0 \le a_1 < b\ ,\ N_1 < N_0.$$

În acelaşi mod, determinăm şirul finit:

$$N > N_0 > N_1 > N_2 .> ... > 0.$$

Deci, există un număr natural n astfel încât

$$N_n = 0$$

sau, echivalent,

$$N_{n-1} = 0 \cdot b + a_n, \text{ cu } 0 < a_n < b.$$

Deci, am obţinut:

$$N = N_0 b + a_0$$
$$N_0 = N_1 b + a_1$$
$$....$$
$$N_{n-2} = N_{n-1}\, b + a_{n-1}$$
$$N_{n-1} = a_n$$

Înmulţind aceste egalităţi cu puteri ale lui b (de la 0 la n) şi adunându-le, obţinem:

$$N = a_n b^n + a_{n-1} b^{n-1} + ...+ a_1 b + a_0.$$

Vom demonstra prin reducere la absurd că numerele $n, a_0, a_1, ..., a_n$ sunt unice. Presupunem prin absurd că există numerele $p, c_0, c_1, ..., c_p$ astfel încât

$$N = c_p b^p + c_{p-1} b^{p-1} + ...+ c_1 b + c_0.$$

Dacă $n < p$ atunci,

$$N = \sum_{i=0}^{n} a_i b^i < b^{n+1} \le b^p \le \sum_{i=0}^{p} c_i b^i = N, \text{ ceea ce este absurd.}$$

Deci, $n \ge p$.

În acelaşi mod, se poate arăta că $p \ge n$. Deci, $n = p$.

Mai trebuie demonstrat că $a_i = c_i$ for $0 \le i \le n$.

Dacă $n = 0$ atunci $a_0 = N = c_0$.

Dacă $n > 0$, atunci

$$N = a_0 + b(a_n b^{n-1} + a_{n-1} b^{n-2} + ...+ a_1),\ 0 \le a_0 < b$$

şi

$$N = c_0 + b(c_n b^{n-1} + c_{n-1} b^{n-2} + ...+ c_1)\ ,\ 0 \le c_0 < b.$$

Din Teorema împărţirii cu rest rezultă că

$a_0 = c_0$ pentru că sunt resturi în împărtirea a două numere întregi

şi

$a_n b^{n\text{-}1} + a_{n\text{-}1} b^{n\text{-}2} + ...+ a_1 = c_n b^{n\text{-}1} + c_{n\text{-}1} b^{n\text{-}2} + ...+ c_1$ pentru că sunt câturi în împărtirea a două numere întregi.

Succesiv, obţinem că $a_1 = c_1$, $a_2 = c_2$, ..., $a_n = c_n$.

Observaţie *a*) Teorema sistemelor de numeraţie arată că există o corespondenţă biunivocă între numerele naturale strict pozitive N şi şirurile finite de numere naturale $a_0, a_1,..., a_n$ unde $0 \le a_i < b$ pentru orice i, $0 \le i \le n$ şi $0 < a_n < b$. Deci, orice număr natural N poate fi scris în oricare dintre următoarele forme:

$$N = a_n a_{n\text{-}1}...a_1 a_0 = a_n b^n + a_{n\text{-}1} b^{n\text{-}1} + ...+ a_1 b + a_0.$$

Suma $a_n b^n + a_{n\text{-}1} b^{n\text{-}1} + ...+ a_1 b + a_0$ se numeşte *notaţie extinsă*.

a_n este *cea mai semnificativă cifră*, iar a_0 este *cea mai puţin semnificativă cifră* a numărului N.

b) Atunci când se lucrează cu diferite baze, se precizează şi baza: vom nota $a_n a_{n\text{-}1}...a_1 a_{0(\text{b})}$ în loc de $a_n a_{n\text{-}1}...a_1 a_0$.

O consecinţă importantă a Teoremei sistemelor de numeraţie este algoritmul sistemelor de numeraţie, care este o metodă de conversie a unui număr întreg din baza 10 într-o bază oarecare *b*.

Algoritm sisteme numeratie;
begin
    read N, b;
    $i \leftarrow 0$;
    repeat
        $a_i \leftarrow N \,\%\, b$; //restul împăţirii
        $N \leftarrow [N / b]$; //câtul împărţirii
        $i \leftarrow i + 1$;
    until $N = 0$;
    write cifrele $a_{i\text{-}1}, a_{i\text{-}2}, ... , a_0$ ale numărului scris în baza b;
end.

Exemplu

Vom converti numărul zecimal 493 în baza 8.

Împărţind 493 la 8 obţinem câtul 61 şi restul 5.
Împărţind 61 la 8 obţinem câtul 7 şi restul 5.
Împărţind 7 la 8 obţinem câtul 0 şi restul 7.

Conversia în baza 8 a numărului zecimal 493 se obţine scriind, în ordine inversă, resturile obţinute la împărţirile succesive la 8. Deci, $493_{(10)} = 755_{(8)}$.

Exemplu
Vom converti numărul zecimal 168 în baza 16.

Împărţind 168 la 16 obţinem câtul 10 şi restul 8.
Împărţind 10 la 16 obţinem câtul 0 şi restul 10, căruia îi corespunde cifra hexazecimală A.
Conversia în baza 16 a numărului 168 se obţine scriind, în ordine inversă, resturile obţinute la împărţirile succesive la 16. Deci, $168_{(10)} = A8_{(16)}$.

M1.U1.5. Rezumat

Există două categorii de sisteme de numeraţie: sisteme de numeraţie poziţionale şi sisteme de numeraţie aditive sau nepoziţionale. Sistemul roman este un sistem de numeraţie nepoziţional. Sistemul zecimal, pe care îl folosim de obicei şi sistemul binar folosit de calculatoare sunt sisteme poziţionale.

Orice număr natural strict pozitiv N poate fi scris în mod unic sub forma

$$\boldsymbol{N = a_n b^n + a_{n-1} b^{n-1} + \ldots + a_1 b + a_0}$$

unde $0 \le a_i < b$ pentru orice i, $0 \le i \le n$ şi $0 < a_n < b$,

sau echivalent $\boldsymbol{N = a_n a_{n-1} \ldots a_1 a_{0(b)}}$.

Algoritmul sistemelor de numeraţie este o metodă de conversie a unui număr întreg din baza 10 într-o bază oarecare $b>1$.

M1.U1.6. Test de evaluare a cunoştinţelor

1. Scrieţi numărul zecimal 123 în baza 2, în baza 5, în baza 8 şi în baza 16.
2. Scrieţi numărul zecimal 58674 în baza 5, în baza 8, în baza 9 şi în baza 16.

Unitatea de învăţare M1.U2. Aritmetica în baza *b*

Cuprins

M1.U2.1. Introducere

Sistemele de numeraţie au apărut din necesitatea oamenilor de a număra şi de a socoti. În zilele noastre, sistemul zecimal se foloseşte aproape pretutindeni, însă este uneori necesar să efectuăm operaţii în alte sisteme de numeraţie. În continuare vom studia aritmetica într-o bază oarecare *b*.

M1.U2.2. Obiectivele unităţii de învăţare

Această unitate de învăţare îşi propune ca obiectiv principal însuşirea de către studenţi a aritmeticii într-o bază oarecare b, $b > 1$.

La sfârşitul acestei unităţi de învăţare studenţii vor fi capabili să:

- Să compare numere scrise în orice bază *b*
- Să efectueze adunări, scăderi şi înmulţiri de numere întregi într-o bază dată *b*.

Durata medie de parcurgere a acestei unităţi de învăţare este de 3 ore.

M1.U2.3. Compararea numerelor în baza *b*

Teoremă (Compararea numerelor naturale) Fie $b \in \mathbf{N}$, $b > 1$ şi fie a şi c două numere naturale scrise în baza b, $a = a_m a_{m-1} \ldots a_1 a_{0(b)}$, $c = c_n c_{n-1} \ldots c_1 c_{0(b)}$. Atunci

$a < c$ dacă şi numai dacă $m < n$ sau

$$m = n \text{ şi } a_p < c_p \text{ unde } p = \max \{ i \mid a_i \neq c_i \}.$$

Demonstraţie. Fie $a < c$.

Presupunem prin absurd că $m > n$.

Atunci $c = \sum_{i=0}^{n} c_i b^i \le \sum_{i=0}^{n} (b-1) b^i = b^{n+1} - 1 < b^{n+1} \le b^m \le a$, ceea ce este absurd.

Deci, $m \le n$.

Dacă $m < n$ atunci stop, altfel înseamnă că $m = n$. În acest caz, evident că $a_p < c_p$, unde $p = \max\ \{i \mid a_i \ne c_i\}$.

Reciproc, dacă $m < n$ atunci $a = \sum_{i=0}^{m} a_i b^i \le \sum_{i=0}^{m} (b-1) b^i = b^{m+1} - 1 < b^{m+1} \le b^n \le c$. Deci, $a < c$.

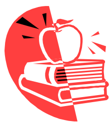
Dacă $m = n$ şi $a_p < c_p$ unde $p = \max\ \{i \mid a_i \ne c_i\}$, atunci

$$c - a = (c_p - a_p)\, b^p + (c_{p-1} b^{p-1} + ... + c_0) - (a_{p-1} b^{p-1} + ... + a_0) >$$
$$> b^p + (c_{p-1} b^{p-1} + ... + c_0) - b^p \ge 0.$$

Deci, $a < c$.

Observaţie Teorema anterioară ne oferă o metodă de comparare a numerelor naturale scrise în baza b. Dacă dorim să comparăm două numere întregi, nu neapărat pozitive, pot apărea 3 cazuri:

1. Amândouă sunt pozitive. În acest caz se aplică teorema anterioară.
2. Unul este negativ şi celălalt pozitiv. Evident, numărul negativ este mai mic decât cel pozitiv.
3. Amândouă sunt negative: $a, c < 0$. În acest caz $a > c$ dacă şi numai dacă $-a < -c$.



Exemple

$101011_{(2)} > 11111_{(2)}$

$AB56_{(16)} > A9FF_{(16)}$

$-1230_{(5)} > -2121_{(5)}$ pentru că $1230_{(5)} < 2121_{(5)}$.

M1.U2.4. Adunarea numerelor în baza *b*

Fie $b \in \mathbf{N}$, $b > 1$ şi fie a şi c două numere naturale scrise în baza b, $a = a_m a_{m-1} ... a_1 a_{0(b)}$, $c = c_n c_{n-1} ... c_1 c_{0(b)}$. Dorim să determinăm suma $s = a + c$.

Avem $a = a_0 + a_1 b + a_2 b^2 + ... + a_m b^m$ şi $c = c_0 + c_1 b + c_2 b^2 + ... + c_n b^n$.

Deoarece $a_0 < b$ şi $c_0 < b$, rezultă că $a_0 + c_0 < 2b$ sau, echivalent,

$$a_0 + c_0 = r_1 \cdot b + s_0,$$

unde $0 \le s_0 < b$, iar r_1 este 0 sau 1 şi se numeşte *transport*.

Mai exact,

dacă $a_0 + c_0 < b$ atunci $s_0 = a_0 + c_0$ şi $r_1 = 0$

dacă $a_0 + c_0 \ge b$ atunci $s_0 = a_0 + c_0 - b$ şi $r_1 = 1$.

Deci,

$$s = a + c = s_0 + (a_1 + c_1 + r_1) b + (a_2 + c_2) b^2 + ...$$

Evident că

$$a_1 + c_1 + r_1 < 2b.$$

Deci,

$$a_1 + c_1 + r_1 = r_2 b + s_1,$$

unde $0 \le s_1 < b$, iar transportul r_2 este 0 sau 1.

În acelaşi fel, determinăm toate cifrele sumei s:

$$s_i = (a_i + c_i + r_i)\%b,\ \forall\ i \ge 0,$$

unde $r_0 = 0$,

$$r_i = \begin{cases} 0, \text{dacă } a_{i\text{-}1} + c_{i\text{-}1} + r_{i\text{-}1} < b \\ 1, \text{altfel} \end{cases},\ \forall\ i > 0.$$

Observaţie 1) Dacă adunăm două numere naturale a şi c, fiecare având $(m+1)$ cifre, suma $s = a + c$ poate avea:

$(m+1)$ cifre dacă $a_m + c_m + r_m < b$ sau

$(m+2)$ cifre dacă $a_m + c_m + r_m \ge b$. În acest caz $s_{m+1} = 1$.

2) Dacă cele două numere naturale a şi c un un număr diferit de cifre atunci:

dacă $m > n$ considerăm $c_{n+1} = c_{n+2} = ... = c_m = 0$

dacă $m < n$ considerăm $a_{m+1} = a_{m+2} = ... = a_n = 0$.

Algoritm adunare;

begin

    read b, m, a_m, $a_{m\text{-}1}$, ..., a_0, n, c_n, $c_{n\text{-}1}$, ..., c_0;

    if $m > n$ **then**

        for $i = n+1$ **to** m **do** $c_i = 0$;

    else

        if $m < n$ **then begin**

                for $i = m+1$ **to** n **do** $a_i = 0$;

                $m \leftarrow n$;

        end;

    $r \leftarrow 0$;

    for $i = 0$ **to** m **do begin**

            $s_i \leftarrow (a_i + c_i + r)\%b$;

            $r \leftarrow (a_i + c_i + r)/b$;

    end;

    if $r = 1$ **then**

        write 1, s_m, $s_{m\text{-}1}$, ... , s_0;

    else

        write s_m, $s_{m\text{-}1}$, ... , s_0;

end.

Exemple

$\begin{array}{r} 101011_{(2)} + \\ \underline{111_{(2)}} \\ 110010_{(2)} \end{array}$	$\begin{array}{r} 12345_{(8)} + \\ \underline{34274_{(8)}} \\ 46641_{(8)} \end{array}$	$\begin{array}{r} A1B2C_{(16)} + \\ \underline{26F54_{(16)}} \\ C8A80_{(8)} \end{array}$

Efectuaţi următoarele adunări:

$101011_{(2)} + 11111_{(2)}$

$AB56_{(16)} + A9FF_{(16)}$

$1230_{(5)} + 2121_{(5)}$.

M1.U2.5. Scăderea numerelor în baza *b*

Fie $b \in \mathbf{N}$, $b > 1$ şi fie a şi c două numere naturale scrise în baza b, $a = a_m a_{m-1} \ldots a_1 a_{0(b)}$, $c = c_n c_{n-1} \ldots c_1 c_{0(b)}$, cu $a \geq c$. Dorim să determinăm diferenţa $d = a - c$.

Se efectuează scăderi în baza 16 atunci când se lucrează direct cu conţinutul locaţiilor de memorie. Există aplicaţii practice în care trebuie să scădem o adresă de memorie dintr-o altă adresă de memorie mai mare.

```
Algoritm scadere;
begin
        read b, m, a_m, a_{m-1}, ..., a_0, n, c_n, c_{n-1}, ..., c_0;
        if m > n then
                for i = n +1 to m do c_i= 0;
        for i = 0 to m do
                if a_i ≥ c_i then
                        d_i ← a_i - c_i ;
                else
                begin
                        se împrumută 1 de la a_{i+1}; // a_{i+1} = a_{i+1} - 1;
                        d_i ← b + a_i - c_i;
                end;
        while d_m = 0 do
                m ← m – 1;
        write d_m, d_{m-1}, ... , d_0;
end.
```

Exemple

$101011_{(2)}$ -	$72345_{(8)}$ -	$A4EC_{(16)}$ -
$\underline{\quad 111}_{(2)}$	$\underline{34274}_{(8)}$	$\underline{34B4}_{(16)}$
$100100_{(2)}$	$36051_{(8)}$	$7038_{(16)}$

Efectuaţi următoarele scăderi:
$101011_{(2)} - 1111_{(2)}$
$AB56_{(16)} - A9FF_{(16)}$
$1230_{(5)} - 221_{(5)}$.

M1.U2.6. Înmulţirea numerelor în baza *b*

Înmulţirea a două numbere *a* şi *c* în baza *b* constă în următoarele operaţii:

1. Înmulţirea numărului *a* cu puteri ale bazei *b*
2. Înmulţirea numărului *a* cu cifre ale numărului *c*
3. Adunări de numere naturale.

<u>Înmulţirea numărului *a* cu o putere a bazei *b*</u>

Fie $a = a_m a_{m-1} \ldots a_1 a_{0(b)}$.
Atunci $a \cdot b^j = (a_0 + a_1 b + a_2 b^2 + \ldots + a_m b^m) \cdot b^j = a_0 b^j + a_1 b^{j+1} + a_2 b^{j+2} + \ldots + a_m b^{m+j}$ $= a_m a_{m-1} \ldots a_0 \underbrace{00\ldots0}_{j \text{ ori}}$.

<u>Înmulţirea numărului *a* cu o cifră a numărului *c*</u>

Fie $a = a_m a_{m-1} \ldots a_1 a_{0(b)} = a_0 + a_1 b + a_2 b^2 + \ldots + a_m b^m$ şi c_j o cifră oarecare a numărului *c*.

Deoarece $a_i < b$ pentru orice $i = 0, \ldots, m$ şi $c_j < b$, rezultă că $a_i \cdot c_j < b^2$. Din Teorema împărţirii cu rest rezultă că există câtul unic $q(a_i, c_j)$ şi restul unic $r(a_i, c_j)$ astfel încât:

$$a_i c_j = b \cdot q(a_i, c_j) + r(a_i, c_j),$$

unde $0 \le r(a_i, c_j) < b$ şi $0 \le q(a_i, c_j) < b$.

Deci $a \cdot c_j = \sum_{i=0}^{m} a_i \cdot c_j \cdot b^i = \sum_{i=0}^{m} (b \cdot q(a_i, c_j) + r(a_i, c_j)) \cdot b^i = \sum_{i=0}^{m} q(a_i, c_j) b^{i+1} + \sum_{i=0}^{m} r(a_i, c_j) b^i$.

Exemple

$101011_{(2)} \cdot$	$123_{(5)} \cdot$	$123_{(8)} \cdot$
$101_{(2)}$	$24_{(5)}$	$24_{(8)}$
----------------	-------	-------
101011	1102	514
101011	301	246
----------------	-------	-------
$11010111_{(2)}$	$4112_{(5)}$	$3174_{(8)}$

Efectuaţi următoarele înmulţiri:
$101011_{(2)} - 1111_{(2)}$
$AB56_{(16)} - A9FF_{(16)}$
$1230_{(5)} - 221_{(5)}$.

Observaţie Împărţirea a două numere în baza *b* constă în operaţii deja studiate: adunări, scăderi şi înmulţiri.

Exemplu

$$
\begin{array}{l|l}
2144_{(5)} & \underline{12_{(5)}} \\
\underline{12} & 132 \\
\ \ 44 & \\
\ \ \underline{41} & \\
\ \ \ \ 34 & \\
\ \ \ \ \underline{24} & \\
\ \ \ \ 10 &
\end{array}
$$

Deci $2144_{(5)} = 12_{(5)} \cdot 132_{(5)} + 10_{(5)}$.

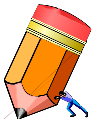
$$
\begin{array}{l|l}
1100101_{(2)} & \underline{101_{(2)}} \\
\underline{101} & 10100 \\
\ \ 10 & \\
\ \ \underline{\ 0} & \\
\ \ 101 & \\
\ \ \underline{101} & \\
\ \ \ \ \ 0 & \\
\ \ \ \ \ \underline{0} & \\
\ \ \ \ \ \ 1 & \\
\ \ \ \ \ \ \underline{0} & \\
\ \ \ \ \ \ \ 1 &
\end{array}
$$

Deci $1100101_{(2)} = 101_{(5)} \cdot 10100_{(2)} + 1_{(2)}$.

Efectuaţi următoarele împărţiri:
$101011_{(2)} : 11_{(2)}$
$1230_{(5)} : 21_{(5)}$
$A085_{(16)} : 1B_{(16)}$.

M1.U2.7. Rezumat

Sistemul de numeraţie pe care îl folosim în marea majoritate a timpului este cel zecimal, însă uneori este necesar să efectuăm calcule şi în alte baze. În acestă unitate de învăţare am arătat cum se efectuează adunările, scăderile, înmulţirile şi împărţirile într-o bază oarecare *b*, *b*>1. De asemenea, am văzut cum pot fi comparate două numere scrise în baza *b*, *b*>1. Ceea ce a putut fi observat cu uşurinţă este faptul că, indiferent de baza în care lucrăm, aplicăm acelaşi algoritm de adunare, de scădere, de înmulţire sau de împărţire.



M1.U2.8. Test de evaluare a cunoştinţelor

Efectuaţi următoarele operaţii:

$110111_{(2)} + 1011_{(2)}$

$110111_{(2)} - 1011_{(2)}$

$110111_{(2)} * 101_{(2)}$

$110111_{(2)} : 11_{(2)}$

$A1B5F_{(16)} + A9F22F_{(16)}$

$A1B5F_{(16)} – BCD1_{(16)}$

$A1B_{(16)} * A2F_{(16)}$

$A1B5F_{(16)} : 1A2_{(16)}$.

Unitatea de învăţare M1.U3. Schimbarea bazei

Cuprins

M1.U3.1. Introducere

Atunci când se lucrează cu mai multe sisteme de numeraţie, apare şi necesitatea de a efectua conversii între diferite baze.

Fie N un număr în baza b, care poate fi dat în oricare din următoarele trei forme echivalente:

$$N = \underbrace{a_n a_{n-1} \ldots a_1 a_0}_{\text{partea intreagă}} . \underbrace{a_{-1} a_{-2} \ldots a_{-m}}_{\text{partea fractională}}$$

$$N = a_n b^n + a_{n-1} b^{n-1} + \ldots + a_1 b + a_0 + a_{-1} b^{-1} + \ldots + a_{-m} b^{-m}$$

$$N = \sum_{i=-m}^{n} a_i b^i$$

Spunem că a_n este *cea mai semnificativă cifră* a numărului N şi că a_{-m} este *cea mai puţin semnificativă cifră* a numărului N.

Observaţie Dacă $m = 0$ atunci N este un număr întreg.

Fie N un număr în baza b, $N = a_n b^n + a_{n-1} b^{n-1} + \ldots + a_1 b + a_0 + a_{-1} b^{-1} + \ldots + a_{-m} b^{-m}$.

Îl vom transforma pe N din baza b în baza h. Există mai multe metode de efectuare a acestei transformări:

1. Metoda substituţiei cu calcule în baza h
2. Metoda substituţiei cu calcule în baza b
3. Metoda substituţiei cu calcule într-o bază intermediară g.

M1.U3.2. Obiectivele unităţii de învăţare

Această unitate de învăţare îşi propune ca obiectiv principal însuşirea de către studenţi a capacităţii de a efectua conversii între diferite baze.

La sfârşitul acestei unităţi de învăţare studenţii vor fi capabili să:

- Să treacă un număr natural dintr-o bază în alta;
- Să treacă un număr raţional dintr-o bază în alta.

Durata medie de parcurgere a acestei unităţi de învăţare este de 3 ore.

M1.U3.3. Metoda substituţiei cu calculele în noua bază

Această metodă constă în scrierea fiecărei cifre a_i a numărului N şi a bazei b în noua bază h. Toate calculele vor fi făcute în noua bază, h.

Vom efectua următoarele conversii:

$a_{i(b)} = c_{i(h)}$, $\forall i$, $-m \le i \le n$

$10_{(b)} = x_{(h)}$.

Deci, $N_{(b)} = (c_n x^n + c_{n-1} x^{n-1} + ... + c_1 x + c_0 + c_{-1} x^{-1} + ... + c_{-m} x^{-m})_{(h)}$.

Exemplu

Pentru trecerea numărului $234_{(5)}$ din baza 5 în baza 2, vom scrie fiecare cifră a numărului şi baza 5 în baza 2:

$2_{(5)} = 10_{(2)}$, $3_{(5)} = 11_{(2)}$, $4_{(5)} = 100_{(2)}$, $10_{(5)} = 101_{(2)}$.

Deci, $234_{(5)} = (2*5^2 + 3*5 + 4)_{(5)} = (10*101^2 + 11*101 + 100)_{(2)} = (10*11001+1111+100)_{(2)} = (110010+1111+100)_{(2)} = 1000101_{(2)}$.

Caz particular

Dacă $b = h^k$ fiecare cifră a numărului scris în baza b va transformată într-un număr în baza h cu k cifre. Apoi aceste numere în baza h sunt juxtapuse.

Exemplu

Pentru trecerea numărului $5436_{(8)}$ din baza 8 în baza 2, vom scrie fiecare cifră a numărului dat în baza 2 pe 3 poziţii, deoarece $8 = 2^3$:

$5_{(8)} = 101_{(2)}$, $4_{(8)} = 100_{(2)}$, $3_{(8)} = 011_{(2)}$, $6_{(8)} = 110_{(2)}$.

Deci, $5436_{(8)} = 101100011110_{(2)}$.

Exemplu

Pentru trecerea numărului A5F.61$_{(16)}$ din baza 16 în baza 2, vom scrie fiecare cifră a numărului hexazecimal în baza 2 pe 4 poziţii, iar apoi le vom concatena.

A$_{(16)}$ = 1010$_{(2)}$, 5$_{(16)}$ = 0101$_{(2)}$, F$_{(16)}$ = 1111$_{(2)}$, 6$_{(16)}$ = 0110$_{(2)}$, 1$_{(16)}$ = 0001$_{(2)}$.

Deci, A5F.61$_{(16)}$ = 101001011111.01100001$_{(2)}$.

Efectuaţi următoarele conversii:

101011$_{(2)}$ = ?$_{(10)}$

1230$_{(5)}$ = ? $_{(2)}$

A085$_{(16)}$ = ? $_{(4)}$

A085$_{(16)}$ = ? $_{(2)}$.

M1.U3.4. Metoda substituţiei cu calculele în vechea bază (metoda împărţirii/înmulţirii bazei)

Metoda substituţiei cu calculele în vechea bază se mai numeşte şi *metoda împărţirii/înmulţirii bazei*. Această metodă este folosită de calculatoare pentru afişarea rezultatelor, baza iniţială fiind 2, iar noua bază fiind 10, de obicei.

Fie $N_{(b)} = a_n a_{n-1} \ldots a_1 a_0 . a_{-1} \ldots a_{-m(b)} = [N]_{(b)} + \{N\}_{(b)}$,

unde $[N]_{(b)}$ este partea întreagă a lui $N_{(b)}$, iar $\{N\}_{(b)}$ este partea sa fracţionară.

Pentru transformarea părţii întregi $[N]_{(b)}$ din baza b în baza h, vom folosi Teorema sistemelor de numeraţie. Partea întreagă a lui $N_{(b)}$, $[N]_{(b)}$, poate fi scrisă în forma:

$$[N]_{(b)} = c_{n'}h^{n'} + c_{n'-1}h^{n'-1} + \ldots + c_1 h + c_0,$$

unde nu se cunosc cifrele $c_{n'}, c_{n'-1}, \ldots, c_1, c_0$, dar pot fi determinate prin împărţiri succesive la h.

$[N]_{(b)} = (\underbrace{c_{n'}h^{n'-1} + c_{n'-1}h^{n'-2} + \ldots + c_1}_{[N_1]_{(b)}})h + c_0$. De aici putem determina c_0, care este restul într-o împărţire a două numere întregi,

$[N_1]_{(b)} = (\underbrace{c_{n'}h^{n'-2} + c_{n'-1}h^{n'-3} + \ldots + c_2}_{[N_2]_{(b)}})h + c_1$, deci putem determina c_1,

.....

$[N_{n'}]_{(b)} = c_{n'}$, deci putem determina $c_{n'}$.

În continuare vom trece partea fracţională $\{N\}_{(b)}$ a lui N din baza b în baza h. Putem scrie $\{N\}_{(b)}$ în forma

$$\{N\}_{(b)} = c_{-1}h^{-1} + c_{-2}h^{-2} + \ldots + c_{-m'}h^{-m'},$$

unde nu se cunosc cifrele $c_{-1}, c_{-2}, \ldots, c_{-m'}$, dar pot fi determinate prin înmulţiri succesive cu h.

$\{N\}_{(b)} h = c_{-1} + \underbrace{c_{-2}h^{-1} + c_{-3}h^{-2} + \ldots + c_{-m'}h^{-m'+1}}_{\{N_1\}_{(b)}}$, deci putem determina c_{-1},

$\{N_1\}_{(b)} h = c_{-2} + \underbrace{c_{-3}h^{-1} + \ldots + c_{-m'}h^{-m'+2}}_{\{N_2\}_{(b)}}$, deci putem determina c_{-2},

....

$\{N_{m'-1}\}_{(b)} h = c_{-m'}$, deci putem determina $c_{-m'}$.

Deci, am obţinut că $N_{(b)} = a_n a_{n-1} \ldots a_1 a_0 . a_{-1} \ldots a_{-m(b)} = c_{n'} c_{n'-1} \ldots c_1 c_0 . c_{-1} \ldots c_{-m'(h)}$.

Exemplu

Trecerea din baza 10 în baza 8 a numărului $234.128_{(10)}$, efectuând calculele în vechea bază, adică în baza 10.

Pentru transformarea părţii întregi, efectuăm împărţiri succesive la 8 şi reţinem resturile în ordinea inversă ordinii în care au fost obţinute:

$234 = 29 * 8 + 2$, rezultă că $c_0 = 2$.
$29 = 3 * 8 + 5$, deci $c_1 = 5$.
$3 = 0 * 8 + 3$, deci $c_2 = 3$.

Deci partea întreagă a numărului dat, $234_{(10)}$, este $352_{(8)}$.

Pentru transformarea părţii fracţionare, efectuăm înmulţiri succesive cu 8 şi reţinem părţile întregi ale produselor în ordinea în care au fost obţinute:

$0.128 * 8 = 1.024$, rezultă că $c_{-1} = 1$.
$0.024 * 8 = 0.192$, deci $c_{-2} = 0$.
$0.192 * 8 = 1.536$, deci $c_{-3} = 1$.

Deci, am obţinut că, $0.128_{(10)} = 0.101_{(8)}$ şi $234.128_{(10)} = 352.101_{(8)}$.

Exemplu

Trecerea din baza 5 în baza 2 a numărului $234.21_{(5)}$, efectuând calculele în vechea bază, adică în baza 5.

Pentru transformarea părţii întregi, efectuăm împărţiri succesive la 2, efectuând calculele în baza 5, şi reţinem resturile în ordinea inversă ordinii în care au fost obţinute:

$234_{(5)} = 114_{(5)} * 2_{(5)} + 1_{(5)}$, rezultă că $c_0 = 1$.
$114_{(5)} = 32_{(5)} * 2_{(5)} + 0_{(5)}$, deci $c_1 = 0$.
$32_{(5)} = 13_{(5)} * 2_{(5)} + 1_{(5)}$, deci $c_2 = 1$.
$13_{(5)} = 4_{(5)} * 2_{(5)} + 0_{(5)}$, rezultă că $c_3 = 0$.
$4_{(5)} = 2_{(5)} * 2_{(5)} + 0_{(5)}$, deci $c_4 = 0$.
$2_{(5)} = 1_{(5)} * 2_{(5)} + 0_{(5)}$, deci $c_5 = 0$.
$1_{(5)} = 0_{(5)} * 2_{(5)} + 1_{(5)}$, deci $c_6 = 1$.

Deci partea întreagă a numărului dat, $234_{(5)}$, este $1000101_{(2)}$.

Pentru transformarea părţii fracţionare, efectuăm înmulţiri succesive cu 2, efectuând calculele în baza 5, şi reţinem părţile întregi ale produselor în ordinea în care au fost obţinute:

$0.21_{(5)} * 2_{(5)} = 0.42_{(5)}$, rezultă că $c_{-1} = 0$.
$0.42_{(5)} * 2_{(5)} = 1.34_{(5)}$, rezultă că $c_{-2} = 1$.
$0.34_{(5)} * 2_{(5)} = 1.23_{(5)}$, rezultă că $c_{-3} = 1$.

Deci, am obţinut că, $0.21_{(5)} = 0.011_{(2)}$ şi $234.21_{(5)} = 1000101.011_{(2)}$.

Caz particular

Dacă $h = b^k$ atunci, pornind de la virgulă către stânga şi către dreapta (dacă numărul are şi parte fracţionară), se formează grupe de câte k cifre. Dacă este necesar, se adaugă zerouri nesemnificative la partea fracţionară astfel încât grupele din dreapta virgulei să aibă toate exact k cifre. Fiecare grup se transformă apoi într-o cifră în baza h. În final, juxtapunem ciferele obţinute în baza h.

Exemple

1) Pentru transformarea numărului binar $1101110111001_{(2)}$ într-un număr în baza 8, vom grupa biţii numărului binar câte 3 de la dreapta la stânga deoarece numărul este întreg, iar $8 = 2^3$:

$$\underbrace{1}\underbrace{101}\underbrace{110}\underbrace{111}\underbrace{001} = \underbrace{001}\underbrace{101}\underbrace{110}\underbrace{111}\underbrace{001}.$$

Apoi transformăm fiecare grup de 3 biţi în cifra corespunzătoare lor din baza 8:

$001_{(2)} = 1_{(8)}$, $101_{(2)} = 5_{(8)}$, $110_{(2)} = 6_{(8)}$, $111_{(2)} = 7_{(8)}$.

Deci, $1101110111001_{(2)} = 15671_{(8)}$.

2) Pentru transformarea numărului binar $10011111.11001_{(2)}$ într-un număr în baza 16, vom grupa biţii numărului binar câte 4 pornind de la punct către stânga şi către dreapta deoarece $16 = 2^4$:

$$\underbrace{1001}\underbrace{1111}.\underbrace{1100}\underbrace{1} = \underbrace{1001}\underbrace{1111}.\underbrace{1100}\underbrace{1000}$$

Apoi transformăm fiecare grup de 4 biţi într-o cifră din baza 16:

$1001_{(2)} = 9_{(16)}$, $1111_{(2)} = F_{(16)}$, $1101_{(2)} = D_{(16)}$, $1000_{(2)} = 8_{(16)}$.

deci, $10011111.11001_{(2)} = 9F.D8_{(16)}$.

M1.U3.5. Metoda substituţiei cu calculele într-o bază intermediară

În general, această metodă este folosită atunci când, folosind creionul şi hârtia, trebuie să transformăm un număr din baza b în baza h, cu b, $h \neq 10$. Deoarece suntem familiarizaţi cu aritmetica în baza 10, preferăm să facem toate calculele în baza 10. Deci, vom folosi baza 10 ca bază intermediară.

Metoda substituţiei cu calcule într-o bază intermediară constă în:

1. Transformarea numărului din baza b în baza g, efectuând calculele în baza g, urmată de
2. Transformarea numărului obţinut din baza g în baza h, efectuând calculele în baza g.

Exemplu

Pentru a transforma numărul $111442_{(5)}$ în baza 8, vom folosi baza 10 ca bază intermediară. Mai întâi transformăm numărul $111442_{(5)}$ în baza 10 astfel:

$$111442_{(5)} = (1*5^5 + 1*5^4 + 1*5^3 + 4*5^2 + 4*5^1 + 2*5^0)_{(10)} = 3997_{(10)}.$$

Apoi, transformăm numărul $3997_{(10)}$ în baza 8:

$3997 = 499*8 + 5$
$499 = 62*8 + 3$
$62 = 7*8 + 6$
$7 = 0*8 + 7$

Deci, $3997_{(10)} = 7635_{(8)}$ şi $111442_{(5)} = 7635_{(8)}$.

Exemplu

Pentru a transforma numărul $208.7_{(9)}$ în baza 7, vom folosi baza 10 ca bază intermediară. Mai întâi transformăm numărul $208.7_{(9)}$ în baza 10 astfel:

$$208.7_{(9)} = (2*9^2 + 0*9^1 + 8*9^0 + 7*9^{-1})_{(10)} = 170.077_{(10)}.$$

Apoi, transformăm numărul $170.077_{(10)}$ în baza 7:

Pentru transformarea părţii întregi, efectuăm împărţiri succesive la 7, efectuând calculele în baza 10, şi reţinem resturile în ordinea inversă ordinii în care au fost obţinute:

$170_{(10)} = 24_{(10)} * 7_{(10)} + 2_{(10)}$, rezultă că $c_0 = 2$.
$24_{(10)} = 3_{(10)} * 7_{(10)} + 3_{(10)}$, deci $c_1 = 3$.
$3_{(10)} = 0_{(10)} * 7_{(10)} + 3_{(10)}$, deci $c_2 = 3$.

Deci partea întreagă a numărului, $170_{(10)}$, este $332_{(7)}$.

Pentru transformarea părţii fracţionare, efectuăm înmulţiri succesive cu 7, efectuând calculele în baza 10, şi reţinem părţile întregi ale produselor în ordinea în care au fost obţinute:

$0.077_{(10)} * 7_{(10)} = 0.539_{(10)}$, rezultă că $c_{-1} = 0$.
$0.539_{(10)} * 7_{(10)} = 3.773_{(10)}$, rezultă că $c_{-2} = 3$.
$0.773_{(10)} * 7_{(10)} = 5.511_{(10)}$, rezultă că $c_{-3} = 5$.

Deci, am obţinut că, $0.077_{(10)} = 0.035_{(7)}$ şi $170.077_{(10)} = 332.035_{(7)}$. În consecinţă, $208.7_{(9)} = 332.035_{(7)}$.

M1.U3.6. Rezumat

Atunci când lucrăm în mai multe sisteme de numeraţie, apare şi necesitatea de a efectua conversii între diferite baze. În această unitate de învăţare, am prezentat trei metode generale de trecere a unui număr oarecare $N = a_n a_{n-1} \ldots a_1 a_0 . a_{-1} \ldots a_{-m}$ din baza b în baza h:

1. Metoda substituţiei cu calcule în baza h
2. Metoda substituţiei cu calcule în baza b
3. Metoda substituţiei cu calcule într-o bază intermediară g.

De asemenea, am arătat că cele mai rapide conversii se pot face între baze care sunt puteri ale aceluiaşi număr.

M1.U3.7. Test de evaluare a cunoştinţelor

Efectuaţi următoarele conversii:

$1485.4_{(10)} = ?_{(2)}$

$110111_{(2)} = ?_{(10)}$

$110111_{(2)} = ?_{(8)}$

$110111_{(2)} =? _{(16)}$

$A1B5F_{(16)} =?_{(10)}$

$A1B_{(16)} =?_{(5)}$

$A1B_{(16)} = ?_{(4)}$

$10101.1_{(2)} = ?_{(8)}$

$10101.1_{(2)} = ?_{(16)}$

$1230_{(5)} = ?_{(2)}$

$A085_{(16)} = ?_{(8)}$.