

Cap 2. Bazele aritmetice ale CN (BA ale CN)

- 2.1. Sisteme de numerație
- 2.2. Conversii din $B_{10} \leftrightarrow B_2, B_8, B_{16}$, $B_2 \leftrightarrow B_8, B_{16}$
- 2.3. Reprezentarea numerelor naturale. Plajă de valori
- 2.4. Reprezentarea numerelor întregi. Plajă de valori
- 2.5. Reprezentarea în VF: CD, CI, CC. Operații
- 2.6. Reprezentarea în VM: SP, DP. Operații
- 2.7. Reprezentarea în BCD. Operații

2.1. Sisteme de numerație

SN= totalitatea $\left. \begin{array}{l} \text{- simbolurilor formate din - litere} \\ \text{- cifre} \end{array} \right\}$ al cărui număr total determină
 $\left. \begin{array}{l} \text{- regulilor de reprezentare a numerelor cu ajutorul simbolurilor} \end{array} \right\}$ BAZA sistemului de numerație

Clasificare

	SN	Baza	Cifre
Poziționale	- sistem binar	B_2	0,1
	- sistem octal	B_8	0,1,2,3,4,5,6,7
	- sistem zecimal	B_{10}	0,1,2,3,4,5,6,7,8,9
	- sistem hexazecimal	B_{16}	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F
Nepoziționale (sistemul roman)			

2.1 (*)

Fiind dat un număr întreg N , se numește **representare în baza b** , orice succesiune de cifre a_n, \dots, a_0 , care satisface următoarele proprietăți:

(1) cifrele lui sunt numere naturale cu proprietatea:

$$0 \leq a_i \leq b-1, \text{ cu } i = \overline{0, n}$$

(2) există egalitatea:

$$N = a_n b^n + a_{n-1} b^{n-1} + \dots + a_0 b^0$$

Când succesiunea de cifre are proprietățile (1) și (2), prin definiție se poate scrie:

$$(N)_b = a_n a_{n-1} \dots a_0$$

2.1 (*)

Deci, reprezentarea unui număr într-o bază b (B_b):

-număr natural: $N = a_n a_{n-1} \dots a_1 a_0;$ $N = \sum_{k=0}^n a_k b^k;$

\downarrow $a_n = \text{p.c.m.s.};$ \swarrow $a_0 = \text{p.c.m.p.s.}$

-număr real: $N = a_n a_{n-1} \dots a_1 a_0 a_{-1} a_{-2} \dots a_{-m};$ $N = \sum_{k=-m}^n a_k b^k;$

\downarrow $a_n = \text{p.c.m.s.};$ \downarrow $a_{-m} = \text{p.c.m.p.s.}$

2.2. Conversii din $B_{10} \leftrightarrow B_2, B_8, B_{16}$, $B_2 \leftrightarrow B_8, B_{16}$

Conversii $B_{10} \rightarrow B_2, B_8, B_{16}$

Exista diferiți algoritmi de conversie a numerelor, pentru $\left\{ \begin{array}{l} \text{partea întregă (P.Î.)} \\ \text{partea fracționară (P.F)} \end{array} \right.$

$$N_b = (N_i)_b + (N_f)_b$$

P.Î= prin \div la bază, se rețin resturile în ordinea LIFO

P.F= prin $*$ la bază, se rețin P.Î în ordinea FIFO

- algoritmul se încheie când se obține 0
- dacă nu se ajunge la 0, există 2 politici de conversie:
 - foarte exactă – la P.F. se vor reține atâtea cifre câte încap în cuvântul calculator;
 - echivalența cantității de informație a nr. B_{10} și a nr. B_2 în care se convertește (1 cifra din B_{10} = cifre în B_2)

Exp: $(23,25)_{10} = (10111,01)_2$

2.2. (*)

Conversii $B_2, B_8, B_{16} \rightarrow B_{10}$

Pentru a converti un număr dintr-o bază de numerație b în zecimală, se va calcula **suma produselor dintre cifra corespunzătoare numărului și baza la puterea specificată de poziția acestuia:**

- pentru P.Î – prima poziție este considerată **poziția 0**, corespunzătoare cifrei c.m.p.s;
- pentru P.F- prima **poziție** este **-1**, corespunzătoare primei cifre care urmează după virgulă.

Puterea: $5 \ 4 \ 3 \ 2 \ 1 \ 0 \ -1 \ -2 \ -3$

Exp: $(1 \ 0 \ 1 \ 0 \ 1 \ 1, \ 1 \ 0 \ 1)_2 = 1*2^5 + 0*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + 0*2^0 + 1*2^{-1} + 0*2^{-2} + 1*2^{-3} = 32 + 8 + 2 + 1 + 1/2 + 1/8 = (43,625)_{10}$

2.2 (*)

Conversii $B_2 \rightarrow B_8, B_{16}$:

- $8 = 2^3, 16 = 2^4$

- se fac grupări de **3**, respectiv **4 cifre binare**

- de la poziția virgulei spre **stânga** pentru P.Î, completându-se cu zero prima parte a numărului (pentru o forma grupa de 3, respectiv 4 cifre),

- de la poziția virgulei spre **dreapta**, pentru P.F, completându-se cu zero ultima parte a numărului (pentru o forma grupa de 3, respectiv 4 cifre);

Exp: $(1\ 110\ 011\ 101, 101\ 1)_2 = (1635,54)_8$

$(10\ 0110\ 1011\ 1100, 0011\ 1101\ 11)_2 = (26BC,3DC)_{16}$

2.2 (*)

Conversii $B_8, B_{16} \rightarrow B_2$:

- fiecare cifră din baza 8 este reprezentată de 3 cifre binare ($8 = 2^3$);

- fiecare cifră din baza 16 este reprezentată de 4 cifre binare ($16 = 2^4$)

Exp: $(347,5)_8 = (011\ 100\ 111, 101)_2$

$(5AF4,3E)_{16} = (0101\ 1010\ 1111\ 0100, 0011\ 1110)_2$

2.3. Reprezentarea numerelor naturale N . Plajă de valori

- reprezentare aritmetică

Pe n poziții binare, unde $n=8, 16, 32, 64$:

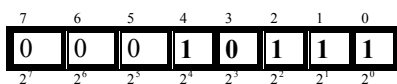


Plaja de valori: $[0, 2^n-1]$

Exp: 23 pe 8 biți

- se reprezintă numărul în baza 2: $(23)_{10} = (10111)_2$

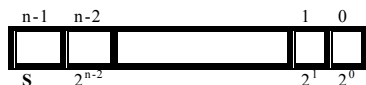
- se introduce numărul în baza 2 în cuvântul calculator, alinierea se face la dreapta, completându-se cu 0 pozițiile libere;



2.4. Reprezentarea numerelor întregi Z . Plajă de valori

-reprezentare algebrică

Pe n poziții binare, unde $n=8, 16, 32, 64$:



$S =$ bitul de semn, $S = \begin{cases} 0, & \text{dacă numărul este pozitiv} \\ 1, & \text{dacă numărul este negativ} \end{cases}$

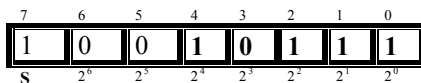
Plaja de valori: $[- 2^{n-1}, 2^{n-1}-1]$

Exp: -23 pe 8 biți

- se reprezintă numărul în baza 2: $(23)_{10} = (10111)_2$

- se introduce numărul în baza 2 în cuvântul calculator, alinierea se face la dreapta, completându-se cu 0 pozițiile libere, până la bitul S de semn;

- se trece valoarea bitului de semn.



2.5. Reprezentarea în VF: CD, CI, CC. Operații

Reprezentarea **numerelor negative** se poate realiza în trei forme:

CD- cod direct- coincide cu reprezentarea algebrică a numerelor întregi, având cifra 1 în poziția semnului;

$$N_{CD} = -1 * 2^{n-1} + \sum_{k=0}^{n-2} a_k 2^k$$

CI- cod invers- se schimbă din CD fiecare cifră binară (din 0 în 1, din 1 în 0); se mai numește și *complementul față de 1*;

$$N_{CI} = -1 * 2^{n-1} + \sum_{k=0}^{n-2} \bar{a}_k 2^k, \text{ unde } \bar{a}_k = 1 - a_k$$

CC- cod complementar – se adună 1 la numărul din CI, pe poziția cea mai puțin semnificativă; se mai numește și *complementul față de 2*;

$$N_{CC} = -1 * 2^{n-1} + \sum_{k=0}^{n-2} \bar{a}_k 2^k + 1 * 2^0$$

CD->CC: se explorează numărul de la dreapta la stânga, toate 0 și primul 1 rămân neschimbate, apoi restul de cifre binare se complementariază față de 1 (0 trece în 1, 1 trece în 0)

Numerelor pozitive au aceeași reprezentare în cele 3 coduri

Conf.Dr.Carmen Timofte

Cap2.Bazele aritmetice ale CN

11

2.5 (*)

Exp: Să se reprezinte +23, -23 pe 8 b, în CD, CI, CC

- se trece numărul în baza 2: $(23)_{10} = (10111)_2$
- se reprezintă pe 8 b

Nr. în B_{10}	Nr în B_2	CD pe 8b	CI pe 8b	CC pe 8b
+23	+10111	0 00 10111	0 00 10111	0 00 10111
-23	-10111	1 00 10111	1 11 01000	1 11 01001

Conf.Dr.Carmen Timofte

Cap2.Bazele aritmetice ale CN

12

2.5. (*)

Operații:

Adunarea:

CD = { se + poziție cu poziție, **FARA** cifra de semn S,
 { dacă apare transport => rezultat INCORECT (depășirea capacității de reprezentare)

CI = { se + poziție cu poziție, **CU** cifra de semn S,
 { dacă apare transport => se + la p.c.p.s. (se adună la poziția cea mai puțin semnificativă)

CC = { se + poziție cu poziție, **CU** cifra de semn S,
 { dacă apare transport => se ignoră

Scăderea se transformă în adunare prin reprezentarea scăzătorului în CI sau CC.

2.5. (*)

Exp: Să se efectueze 25-17 în CD, CI, CC pe 8 biți.

Nr. în B_{10}	Nr în B_2	CD pe 8b	CI pe 8b	CC pe 8b
+25	+11001	0 00 11001	0 00 11001	0 00 11001
-17	- 10001	1 00 10001	1 11 01110	1 11 01111
+25-17		----- (trebuie să fie ambele pozitive)	0 00 00111 + (transport)1 <hr/> 0 00 01000	0 00 01000 1 (transportul se neglijenza)
Verificare			$2^3=8$	$2^3=8$

2.6. Reprezentarea în VM: SP,DP. Operații

Pentru reprezentarea numerelor REALE.

Utilizată la calcule științifice

Un număr N se scrie în VM astfel:

$$N_{VM} = \pm (1,f)_b * b^E$$

unde: b= baza sistemul de numerație (baza 2 de obicei)

E = exponent real

1,f= mantisa normalizată: $1 \leq |1,f| < 2$

Exp: $5 = (101)_2 = 1,01 * 2^2$

2.6 (*)

În calculator se reprezintă în **3 câmpuri**:

-bit de semn S $\begin{cases} 0, & \text{dacă } N \geq 0 \\ 1, & \text{dacă } N < 0 \end{cases}$

- zonă exponent sau caracteristică: EXP = E + constantă

- zona fracției sau mantisa

În calculator se reprezintă în **2 formate**, cu ajutorul câmpurilor de mai sus:

- **SP – simplă precizie** – pe 32 de biți, unde EXP= E+127

31	30	...	23	22	...	0
S	E ₇	←	E ₀	f ₁	→	f ₂₃
semn	exponent			fracție		

- **DP – dublă precizie** - pe 6 de biți, unde EXP= E+1023

63	62	...	52	51	...	0
S	E ₁₀	←	E ₀	f ₁	→	f ₅₂
semn	exponent			fracție		

2.6 (*)

Exp: Să se reprezinte numărul +5 în VM SP

$$+5 = +(101)_2 = +1,01 * 2^2$$

$$EXP = 2 + 127 = (129)_{10} = (83)_{16} = (1000\ 0011)_2$$

31	30	...						23	22			...	0				
0	1	0	0	0	0	0	0	1	1	0	1	0	...	0			
semn									exponent					fracție			

2.6. (*)

Operații de adunare și scădere în VM:

- se compară cei doi exponenți, pentru a-l determina pe cel mai mare;
- se aliniaza mantisa numărului cu exponent mai mic, prin deplasarea virgulei corespunzător exponentului mai mare;
- se adună/scad mantisele aliniate, atribuid exponentul comun;
- se normalizează mantisa, eventual, concomitent cu modificarea exponentului.

Exp.: Să se efectueze în VM: $\frac{3}{4} + 7$

$$\frac{3}{4} = (2^1 + 2^0)/2^2 = 2^{-1} + 2^{-2} = (0,11)_2 = 1,1 * 2^{-1} = 0,0011 * 2^2$$

$$7 = (111)_2 = 1,11 * 2^2$$

$$\frac{3}{4} + 7 = 0,0011 * 2^2 + 1,11 * 2^2 = 1,1111 * 2^2 = (111,11)_2 = 7,75$$

2.7.Reprezentarea în BCD. Operații

Reprezentarea în zecimal codificat binar (8241) se realizează prin exprimarea fiecărei cifre zecimale într-o tetradă binară.

Reprezentarea se realizează în grupuri de 8 cifre binare, în 2 forme:

- Împachetată – în care pe 8 cifre binare se reprezintă câte 2 cifre zecimale exprimate în binar;

$$\begin{array}{|cccc|} \hline 0 & 3 & 4 & 2 \\ \hline 0000 & 0011 & 0100 & 0010 \\ \hline \end{array}$$

- Despachetată - în care pe 8 cifre binare se reprezintă câte 1 cifră zecimală exprimată în binar, precedate de M –marcă- pe cifre binare;

$$\begin{array}{|c|c|c|} \hline & 3 & 4 & 2 \\ \hline | & M & 0011 & | & M & 0100 & | & M & 0010 & | \\ \hline \end{array}$$

2.7. (*)

Adunarea în BCD (8421):

- se exprimă fiecare cifră zecimală printr-o tetradă binară;
- se adună poziție cu poziție, de la dreapta spre stânga, pe fiecare grup de cifre.

Pot apare următoarele situații:

- o rezultat corect,
 - dacă $0000 \leq c \leq 1001$ (adică între 0 și 9), nu se aplică corecții;
- o rezultat incorect, când **se aplică corecții**:
 - dacă $1010 \leq c \leq 1111$ (adică între 10 și 15), atunci se adună 0110 (adică 6), iar transportul generat se adună la suna din poziția următoare;
 - dacă $0000 \leq c \leq 1001$ (adică între 0 și 9), dar a apărut transport la următoarea grupă binară, atunci se adună 0110 (adică 6)

Exp.: Să se efectueze în BCD $683 + 2794$.

$$\begin{array}{cccc} & 1 & 1111 & \\ & 0110 & 1000 & 0011 & + \\ 0010 & 0111 & 1001 & 0100 & \\ \hline 0011 & 1110 & 0001 & 0111 & \\ \hline \cup & \cup & \cup & \cup & \\ 3 & 0100 & 0111 & 7 & \\ \cup & \cup & & & \\ 4 & 7 & & & \end{array}$$