

Laborator 1 - Suport teoretic

Conversii și operații în diferite baze de numerație

Se va studia:

- conversia numerelor întregi și zecimale din baza 10 într-o bază oarecare, în special baza 16 și 2
 - conversia inversă, dintr-o bază oarecare în baza 10, mai ales din baza 16 și 2 în baza 10.
 - conversia din baza 16 direct în baza 2 și invers.
-

Considerații teoretice

Un sistem de numerație este constituit din totalitatea regulilor de reprezentare a numerelor cu ajutorul anumitor simboluri denumite cifre. Pentru orice sistem de numerație, numărul semnelor distincte pentru cifrele sistemului este egal cu baza (b). Deci pentru baza $b=2$ (numere scrise în binar) semnele vor fi cifrele 0 și 1. Pentru baza $b=16$ (hexazecimal) semnele vor fi 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F. Se observă că pentru numerele scrise într-o bază mai mare decât baza 10 (zecimal) se folosesc și alte simboluri (litere) pe lângă cifrele obișnuite din baza 10. Astfel, în cazul numerelor scrise în hexazecimal, literele A,B,C,D,E,F au ca și valori asociate 10,11,12,13,14,15. Moduri de notare: scrierea la sfârșitul numărului în paranteză a bazei, de exemplu: $100101001_{(2)}$, sau $17A6B_{(16)}$.

Conversia numerelor din baza 10 într-o bază oarecare

Algoritmul cel mai simplu constă în împărțirea succesivă a numărului scris în baza 10 la baza spre care se dorește conversia (se împarte numărul la bază, iar în continuare se împarte câtul obținut la bază ș.a.m.d. până când câtul devine 0), după care se iau resturile obținute în ordine inversă, care constituie valoarea numărului în baza cerută.

Exemple:

1. Să se convertească 347 din baza 10 în baza $16_{(16)}$. Mai întâi se convertește în baza 16 pentru că aceasta se realizează prin mai puține împărțiri decât conversia în baza 2.
-

$$\begin{array}{r|l}
 347 & 16 \\
 \hline
 32 & 21 & 16 \\
 \hline
 27 & 16 & 1 & 16 \\
 \hline
 16 & 5 & 0 & 0 \\
 \hline
 11 & 1 & &
 \end{array}$$

(adică "B")

Astfel, luând resturile în ordine inversă obținem $15B_{(H)}$.

$$347_{(D)} = 15B_{(H)}$$

2. Să se convertească 57 din baza 10 în baza $2_{(B)}$.

$$\begin{array}{r|l}
 57 & 2 \\
 \hline
 56 & 28 & 2 \\
 \hline
 1 & 28 & 14 & 2 \\
 & 0 & 14 & 7 & 2 \\
 & & 0 & 6 & 3 & 2 \\
 & & & 1 & 2 & 1 \\
 & & & & 1 &
 \end{array}$$

$$57_{(D)} = 111001_{(B)}$$

Există și o metodă mai rapidă de conversie a numerelor între bazele 2 și 16 ținând cont că pentru fiecare cifră hexa există 4 cifre binare corespundente:

Valoarea în zecimal	Valoarea în hexazecimal	Numărul binar coresp. cifrei hexa
0	0	0000

1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Mai trebuie ținut cont la trecerea unui număr prin bazele 2 și 16 că gruparea cifrelor din baza 2 se face de la dreapta la stânga (prin completare cu zerouri la stânga numărului dacă este cazul, deci în partea care nu-i afectează valoarea).

Exemple:

1. Să se convertească 347 din baza 10 în bazele 2 și 16.

$$347_{(D)} = 15B_{(H)} = 0001\ 0101\ 1011_{(B)}$$

Alte exemple:

```

2 = 2(10) = 10(2)
62(10) = 111110(2)
1995(10) = 11111001011(2)
1024(10) = 1000000000(2)

```

Conversia unui număr dintr-o bază oarecare în baza 10

Pentru a converti un număr dintr-o bază oarecare în baza 10 se poate folosi formula definită în prima parte a lucrării și anume:

Nr (b) = C_n C_{n-1} C_{n-2} ... C₂C₁ C₀

atunci valoarea sa în baza 10 va fi:

Nr (10) = C_n * bⁿ + C_{n-1} * bⁿ⁻¹ + ... + C₂ * b² + C₁ * b¹ + C₀

Exemple:

1. Se dă numărul întreg în hexazecimal 3A8_(H) și se cere valoarea sa în zecimal:

$$N = 3 \cdot 16^2 + 10 \cdot 16^1 + 8 = 3 \cdot 256 + 160 + 8 = 936_{(10)}$$

2. Se dă numărul întreg în hexazecimal 86C_(H) și se cere valoarea sa în zecimal:

$$86C_{(16)} = 8 \cdot 16^2 + 6 \cdot 16 + 12 = 2156_{(10)}$$

3. Se dă numărul întreg în binar 1101101₍₂₎ și se cere valoarea sa în zecimal

$$1101101_{(2)} = 1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2 + 1 = 109_{(10)}$$

Alte exemple:

```

1010011(2) = 83(10)
11100011(2) = 227(10)
1000000000(2) = 512(10)
11001(2) = 25(10)

```

Instrumente de lucru pentru laborator

Instrumentele de lucru pentru laborator sunt următoarele:

- Editor: Notepad++
- Asamblor: NASM
- Linker: ALINK
- Debugger: Olly DBG

Setul de instrumente poate fi descarcat de aici: [ASM tools](#). Studenții care folosesc alte sisteme de operare decât Windows pot utiliza fie un emulator (tool-urile au fost testate cu succes cu Wine), fie pot rula tool-urile într-o mașină virtuală de Windows. Pentru a ușura redactarea programelor ASM, setul de instrumente include un editor vizual (Notepad++) care are integrat un plugin ce permite efectuarea majorității operațiilor din câteva combinații de taste.

Instructiuni de utilizare:

- Descarcati setul de instrumente si extrageti continutul arhivei
- Porniti editorul Notepad++ din directorul *npp* (nu utilizati versiunea proprie a editorului)
- Procedati la redactarea programelor ce rezolva problemele care va vor fi comunicate. Puteti utiliza urmatoarele combinatiile de taste ce se regasesc in meniul Plugins -> ASM Plugin:

ASM Code Template	Ctrl+Shift+N	Completeaza in editor un program minimal in ASM
Build ASM	Ctrl+F7	Asambleaza programul curent
Run program	Ctrl+F6	Ruleaza programul curent
Debug program	F6	Depaneaza programul curent, folosind Olly Debugger (documentatia pentru debugger se regaseste in directorul <i>ollydbg</i> din arhiva)

Un exemplu minimal de program în limbaj de asamblare

Exemplu:

```
bits 32 ;asamblare și compilare pentru arhitectura de 32 biți
; definim punctul de intrare in programul principal
global start

; declaram functiile externe necesare programului nostru
extern exit ; indicam asamblorului ca exit exista, chiar daca noi nu o vom defini
import exit msvcrt.dll ; exit este o functie care incheie procesul, este definita in msvcrt.dll
; msvcrt.dll contine exit, printf si toate celelalte functii C-runtime importante
```

```
; segmentul de date in care se vor defini variabilele
segment data use32 class=data
; ...

; segmentul de cod
segment code use32 class=code
start:
; ...

    ; exit(0)
    push dword 0 ; se pune pe stiva parametrul functiei exit
    call [exit] ; apelul functiei exit pentru terminarea executiei programului
```

Laborator 1 - Probleme propuse

Rezolvați pe hârtie și explicați următoarele exerciții:

1. Converteți din baza 10 în 2 și apoi în 16 următoarele numere:

4

10

15

32

2. Converteți din baza 10 în 16 și apoi în 2 următoarele numere:

3

11

16

17

3. Converteți din baza 2 în baza 16 următoarele numere:

1010

0111

1111

10001010

110101111

4. Converteți din baza 16 în baza 2 următoarele numere:

3

A

F

2B

2F8

5. Efectuați următoarele operații în baza 2 (fără a converti în baza 10):

$$1+1$$

$$10+10$$

$$111+1$$

$$1010-1$$

$$1000-10$$

6. Efectuați următoarele operații în baza 16 (fără a converti în baza 10):

$$9+1$$

$$B+2$$

$$F+1$$

$$10+A$$

$$10-2$$

$$B-3$$