



5. Fejezet : Lebegőpontos számok

**The Architecture of Computer Hardware
and Systems Software:
An Information Technology Approach**

**3. kiadás, Irv Englander
John Wiley and Sons Ó2003**

Wilson Wong, Bentley College
Linda Senne, Bentley College



Lebegőpontos számok

- § Valós számok
- § Akkor használjuk a számítógépeknél,
ha
 - § Kívül esik a számítógép integer határán
(túl kicsi vagy túl nagy)
 - § Tizedes törtet tartalmaz



Exponenciális jelölés

§ Hívják még *tudományos jelölésnek*

§ 12345

§ 12345×10^0

§ 0.12345×10^5

§ 123450000×10^{-4}

§ 4 alkotóelem kell egy számhoz

1. Előjel (pl.: "+")
2. Érték vagy *mantissza* (pl.: 12345)
3. Kitevő előjele (pl.: "+" a 10^5 -ben)
4. Kitevő értéke (pl.: 5)

§ Plusz

5. Kitevő alapja (pl.: 10)
6. Tizedesvessző helye (vagy más bázisé) alappont

Fejezet: 5. - Lebegőpontos
számok

5-3



Szabályok összegzése

Mantissza előjele

Kitevő előjele

Tizedes-
vessző
helye

-0.35790

Mantissza

$\times 10^{-6}$

Alap

Kitevő

Fejezet: 5. - Lebegőpontos
számok

5-4



Formai leírás

- § Előre definiált forma, általában 8 bit
- § Megnövelt értékhatár (két jegyű kitevő) kisebb pontosság árán (ötjegyű mantissza)

Mantissza előjele

SEEMMMMM

2 jegyű kitevő 5 jegyű mantissza

Fejezet: 5. - Lebegőpontos számok

5-5



Forma

- § mantissza: előjel karakter az előjel-érték formában
- § Tizedespont a mantissza elején helyezkedik el
- § Excess-N jelölés: komplement ábrázolás
 - § Vegyük az eltolást a középpértéknek, ahol N a középpérték

Ábrázolás	0	49	50	99
Ábrázolt kitevő	-50	-1	0	49

- Növekvő érték +

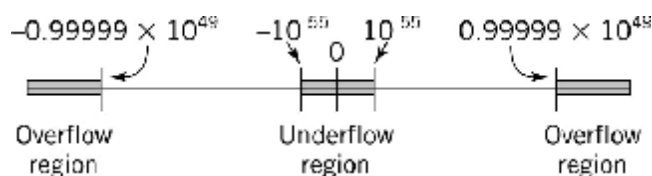
Fejezet: 5. - Lebegőpontos számok

5-6



Túlcsordulás és alulcsordulás

§ Akkor fordul elő, ha az adott számábrázolást tekintve túl kicsi vagy túl nagy számokat szeretnénk kezelni



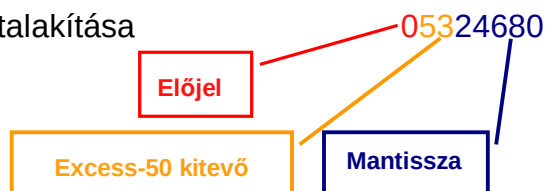
Fejezet: 5. - Lebegőpontos számok

5-7



Példa 1: 246.8035

1. Adjunk kitevőt 246.8035×10^0
2. Pozícionáljuk a tizedesvesszőt a normál alakhoz $.2468035 \times 10^3$
3. Vágjuk le 5 jegyre $.24680 \times 10^3$
4. Szám átalakítása



Fejezet: 5. - Lebegőpontos számok

5-8



Példa 2: 1255×10^{-3}

- | | |
|-----------------------------------|-------------------------|
| 1. Már exponenciális formájú | 1255×10^{-3} |
| 2. Pozícionáljuk a tizedesvesszőt | $.1255 \times 10^{+1}$ |
| 3. Normál alakú | |
| 4. Adjunk hozzá 0-t az 5. jegyhez | $0.1255 \times 10^{+1}$ |
| 5. Szám átalakítása | 05112550 |

Fejezet: 5. - Lebegőpontos számok

5-9



Példa 3: - 0.00000075

- | | |
|-------------------------------|----------------------------|
| 1. Exponenciális jelölés | $- 0.00000075 \times 10^0$ |
| 2. Tizedespont a helyén | |
| 3. Normál alakra hozás | $- 0.75 \times 10^{-6}$ |
| 4. 0 hozzáadása az 5. jegyhez | $- 0.75000 \times 10^{-6}$ |
| 5. Szám átalakítása | 54475000 |

Fejezet: 5. - Lebegőpontos számok

5-10



Átalakítás példák

$$05324567 = 0.24567 \times 10^3 = 245.67$$

$$54810000 = -0.10000 \times 10^{-2} = -0.0010000$$

$$55555555 = -0.55555 \times 10^5 = -55555$$

$$04925000 = 0.25000 \times 10^{-1} = 0.025000$$

Fejezet: 5. - Lebegőpontos
számok

5-11



Normál alakra hozás

- § Toljuk el balra a számokat növelve a kitevőt, míg a kezdő nullák el nem tűnnek
- § Decimális számok konvertálása általános formába
 1. Lássuk el számmal a kitevőt (0 ha még nincs meghatározva)
 2. Növeljük/csökkentsük a kitevő értékét a tizedes pont eltolásával a helyes irányba
 3. Csökkentsük a kitevőt, hogy eltüntessük a nullákat a mantisszából
 4. Javítsunk a pontosságon 0-kat írva vagy elhagyva/kerekítve a kevésbé fontos számjegyeknél

Fejezet: 5. - Lebegőpontos
számok

5-12



Lebegőpontos számolások

§ Összeadás és kivonás

- § A kitevőt és a mantisszát külön kell kezelni
- § A számok kitevőjének egyeznie kell
 - Tizedesvesszők rendezése
 - Kevésbé fontos számjegyek eltűnhetnek
- § mantissza túlcsordulása a kitevő ismételt jobbra tolása szükséges

Fejezet: 5. - Lebegőpontos számok

5-13



Összeadás és kivonás

Két lebegőpontos szám összeadása	05199520 $+ 04967850$
Kitevők rendezése	05199520 0510067850
Mantisszák összeadása 1 átvitelt jelez, ezért	$(1)0019850$
az jobbra tolást igényel	$05210019(850)$
Kerekítés	05210020
Eredmény ellenőrzése	$05199520 = 0.99520 \times 10^1 = 9.9520$ $04967850 = 0.67850 \times 10^{-1} = 0.06785$ $= 10.01985$ $= 0.1001985 \times 10^2$
Exponenciális formában	

Fejezet: 5. - Lebegőpontos számok

5-14



Szorzás és osztás

§ Mantisszák: szorozva vagy osztva

§ Kitevők: összeadva vagy kivonva

§ Normál alak fontos

- A tizedes vessző helyének visszaállításához
- Az eredmény pontosságának fenntartásához

§ Excess értékének igazítása összeadás után

- Példa: mindkét szám kitevője: 3, ábrázolva excess-50 jelöléssel:
 - $53 + 53 = 106$
 - összeadás után kivonás: $106 - 50 = 56$

Fejezet: 5. - Lebegőpontos számok

5-15



Szorzás és osztás

§ Precízió fenntartása:

§ Szorzás normál alakra hozással és kerekítéssel

- Két szám szorzása

05220000
x 04712500
- Kitevők összeadása, offset kivonása

$52 + 47 - 50 = 49$

- mantissza szorzása

$0.20000 \times 0.12500 =$
0.025000000
- Eredmény normál alakra hozása

04825000

- Kerekítés

04825000

- Eredmény ellenőrzése

$05220000 = 0.20000 \times 10^2$
$04712500 = 0.125 \times 10^{-3}$
$= 0.0250000000 \times 10^{-1}$
$= 0.25000 \times 10^{-2}$
- Normál alakra hozás és kerekítés

Fejezet: 5. - Lebegőpontos számok

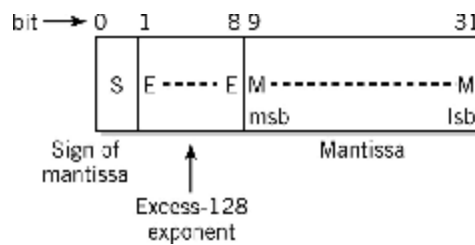
5-16



Lebegőpont a számítógépben

§ Tipikus lebegőpontos forma

- § 32 biten a mérete $\sim 10^{-38}$ -tól 10^{+38} -ig terjed
- § 8 bites kitevő = 256 szint
 - Excess-128 jelölés
- § 23/24 bites mantissza: körülbelül 7 decimális jegy pontosságú



Fejezet: 5. - Lebegőpontos számok

5-17



Lebegőpont a számítógépben

Excess-128 kitevő

	mantissza előjele	mantissza
0	1000 0001	1100 1100 0000 0000 0000 000
		+1.1001 1000 0000 0000 00
1	1000 0100	1000 0111 1000 0000 0000 000
		-1000.0111 1000 0000 0000 000
1	0111 1110	1010 1010 1010 1010 1010 101
		-0.0010 1010 1010 1010 1010 1

Fejezet: 5. - Lebegőpontos számok

5-18



IEEE 754 szabvány

Pontosság	Egyszerű (32 bit)	Dupla (64 bit)
Előjel	1 bit	1 bit
Kitevő	8 bit	11 bit
Jelölés	Excess-127	Excess-1023
Mutató alap	2	2
Terjedelem	2^{-126} -tól 2^{127} -ig	2^{-1022} -tól 2^{1023} -ig
Mantissza	23	52
Decimális pontosság	≈ 7	≈ 15
Értékhatár	$\approx 10^{-45}$ -tól 10^{38} -ig	$\approx 10^{-300}$ -tól 10^{300} -ig

Fejezet: 5. - Lebegőpontos
számok

5-19



IEEE 754 szabvány

§ 32 bit-es lebegő pontos érték definíció

Kitevő	mantissza	Érték
0	± 0	0
0	Nem 0	$\pm 2^{-126} \times 0.M$
1-254	Bármilyen	$\pm 2^{(E-127)} \times 1.M$
255	± 0	$\pm \infty$
255	Nem 0	Speciális eset

Fejezet: 5. - Lebegőpontos
számok

5-20



Átalakítás: 10-es és 2-es alap

§ Két lépés

- § A számok egész és tört részét külön kell átalakítani, beágyazott tizedes vagy bináris ponttal
- § Az átalakítás végrehajtása előtt az exponenciális alakban lévő számokat vissza kell alakítani egyszerű decimális vagy bináris kevert számmá vagy törtté

Fejezet: 5. - Lebegőpontos számok

5-21



Átalakítás: 10-es és 2-es alap

§ 253.75_{10} átalakítása bináris lebegőpontos formába

§ Szám szorzása 100-zal 25375

§ Binárisra alakítás 110 0011 0001 1111 vagy $1.1000\ 1100\ 0111\ 11 \times 2^{14}$

§ IEEE ábrázolás 0 1000 1101 10 0011 0001 1111

Előjel

Excess-127 kitevő =
 $127 + 14$

mantissza

§ Az eredeti decimális értéket visszkapjuk, ha elvégzünk egy 100_{10} -zal való osztást.

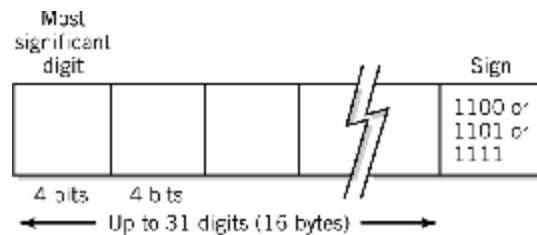
Fejezet: 5. - Lebegőpontos számok

5-22



Tömörített decimális forma

- § Dollárok és centek ábrázolása valós számokkal
- § A vállalat-orientált nyelvek támogatják, mint a COBOL
- § IBM 370/390 és Compaq Alpha



Fejezet: 5. - Lebegőpontos számok

5-23



Programozási megfontolás

- § Integer előnyei
 - § A számítógép könnyebben használja
 - § Precízebb
 - § Gyorsabb
 - § Kisebb tárhely és idő igény
- § A legtöbb magas szintű nyelv kettő vagy több formát szolgáltat
 - § Rövid egész (16 bit)
 - § Hosszú egész (64 bit)

Fejezet: 5. - Lebegőpontos számok

5-24



Programozási megfontolás

§ Valós számok

- § Változónak vagy konstansnak van tört része
- § A nagyon kicsi vagy nagyon nagy értékű számok kívül esnek az integer ábrázolási határán
- § A program lehet, hogy nem lesz elég precíz a feladat végrehajtásához
- § Tömörített decimális számábrázolás vonzóbb a vállalati alkalmazásoknál