

Adatstruktúrák és Algoritmusok

1. gyakorlat

Lebegőpontos számábrázolás

A tanult függvények átisméltése

Alsó egészrész függvény

$$\lfloor \cdot \rfloor : \mathbb{R} \rightarrow \mathbb{Z}$$

$$\lfloor x \rfloor := \max\{z \in \mathbb{Z} \mid z \leq x\} \quad (x \in \mathbb{R}),$$

tehát az alsó egészrész függvény minden valós számhoz hozzárendeli a számnál kisebb vagy egyenlő egészek közül a legnagyobbat.

Feladat Határozza meg a $\lfloor 7.13 \rfloor$ és az $\lfloor -7.13 \rfloor$ értékét.

Mo.:

$$\lfloor 7.13 \rfloor = 7,$$

$$\lfloor -7.13 \rfloor = -8.$$

Felső egészrész függvény

$$[\cdot] : \mathbb{R} \rightarrow \mathbb{Z}$$

$$[x] := \max\{z \in \mathbb{Z} \mid x \leq z\} \quad (x \in \mathbb{R}),$$

tehát a felsőegészrész függvény minden valós számhoz hozzárendeli a számnál nagyobb vagy egyenlő egészek közül a legkisebbet.

Feladat Határozza meg a $[7.13]$ és $[-7.13]$ értékét.

Mo.:

$$[7.13] = 8,$$

$$[-7.13] = -7.$$

Kerekítőfüggvény

$\text{round} : \mathbb{R} \rightarrow \mathbb{Z}$

$$\text{round}(x) := \lfloor x + 0.5 \rfloor \quad (x \in \mathbb{R}).$$

Feladat *Határozza meg a $\text{round}(7.13)$, $\text{round}(7.56)$, $\text{round}(-7.13)$, $\text{round}(-7.56)$ értékét.*

Mo.:

$$\text{round}(7.13) = \lfloor 7.63 \rfloor = 7,$$

$$\text{round}(7.56) = \lfloor 8.06 \rfloor = 8,$$

$$\text{round}(-7.13) = \lfloor -6.63 \rfloor = -7,$$

$$\text{round}(-7.56) = \lfloor -7.06 \rfloor = -8.$$

Törtrészfüggvény

$$\{\cdot\} : \mathbb{R} \rightarrow [0, 1[$$

$$\{x\} := x - \lfloor x \rfloor \quad (x \in \mathbb{R})$$

Feladat Határozza meg az $\{7.13\}$, $\{-7.13\}$ értékeket.

Mo.:

$$\{7.13\} = 7.13 - \lfloor 7.13 \rfloor = 7.13 - 7 = 0.13,$$

$$\{-7.13\} = -7.13 - \lfloor -7.13 \rfloor = -7.13 - (-8) = 0.87.$$

A div függvény

$$\text{div} : \mathbb{R} \times (\mathbb{R} \setminus \{0\}) \rightarrow \mathbb{Z}$$

$$\text{div}(a, b) := a \text{ div } b := \left\lfloor \frac{a}{b} \right\rfloor \quad (a \in \mathbb{R}, b \in \mathbb{R} \setminus \{0\}).$$

Feladat Határozza meg a $13 \text{ div } 5$ és a $(-13) \text{ div } 5$ $13 \text{ div } (-5)$ és a $(-13) \text{ div } (-5)$ értékét.

Mo.:

$$13 \text{ div } 5 = \left\lfloor \frac{13}{5} \right\rfloor = \lfloor 2.6 \rfloor = 2,$$

$$(-13) \text{ div } 5 = \left\lfloor -\frac{13}{5} \right\rfloor = \lfloor -2.6 \rfloor = -3,$$

$$13 \text{ div } (-5) = \left\lfloor \frac{13}{-5} \right\rfloor = \lfloor -2.6 \rfloor = -3,$$

$$(-13) \text{ div } (-5) = \left\lfloor -\frac{-13}{-5} \right\rfloor = \lfloor 2.6 \rfloor = 2.$$

A mod függvény

$$\text{mod}(\cdot, \cdot) : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$$

$$\text{mod}(a, b) := a \text{ mod } b := \begin{cases} a, & \text{ha } b = 0, \\ a - b \cdot \left\lfloor \frac{a}{b} \right\rfloor, & \text{ha } b \neq 0. \end{cases}$$

Feladat Számoljuk, ki $a \text{ mod } 5$, $(-13) \text{ mod } 5$, $13 \text{ mod } (-5)$, $(-13) \text{ mod } (-5)$ értékeket.

Mo.:

$$13 \text{ mod } 5 = 13 - 5 \cdot \left\lfloor \frac{13}{5} \right\rfloor = 13 - 5 \cdot 2 = 3,$$

$$(-13) \text{ mod } 5 = -13 - 5 \cdot \left\lfloor -\frac{13}{5} \right\rfloor = -13 - 5 \cdot (-3) = 2,$$

$$13 \text{ mod } (-5) = 13 - (-5) \cdot \left\lfloor -\frac{13}{5} \right\rfloor = 13 - (-5) \cdot (-3) = -2,$$

$$(-13) \text{ mod } (-5) = -13 - (-5) \cdot \left\lfloor \frac{13}{5} \right\rfloor = -13 - (-5) \cdot 2 = -3.$$

A maradékos osztás tétele (Euklidész)

Ha $a, b \in \mathbb{R}$, $b \neq 0$, akkor léteznek olyan $q, r \in \mathbb{Z}$ számok, amelyekkel

$$a = qb + r, \quad \text{ahol} \quad a \leq |r| < |b|$$

Bizonyítás $q = a \operatorname{div} b$, $r = a \operatorname{mod} b$ megfelelőek.

Elnevezések:

- az a számot **osztandónak**,
- a b számot **osztónak**,
- a q számot **hányadosnak**,
- az r számot **maradéknak** nevezzük.

Az $a \in \mathbb{R}$ szám felírása a $b \in \mathbb{Z}_+$ alapú számrendszerben

Egy tetszőleges a pozitív valós szám rögzített b alapszám esetén egyértelműen írható fel

$$a = c_n \dots c_0 \cdot d_1 d_2 \dots_{(b)} \doteq c_n b^n + \dots + c_0 b^0 + d_1 b^{-1} + d_2 b^{-2} + \dots$$

A felírás egyértelműségéhez szükséges, hogy nem engedjük meg, hogy valahonnan kezdve csupa $(b - 1)$ jegyek álljanak.

Például a 10-es számrendszerben nem állhatnak valahonnan kezdve csupa 9-esek, a 2-es számrendszerben nem állhatnak valahonnan kezdve csupa 1-esek, illetve a 16-os számrendszerben nem állhatnak valahonnan kezdve csupa F jegyek.

Ennek oka, hogy például a 10-es számrendszerben, a mértani sor összegképlete alapján kapjuk, hogy

$$0.99999 \dots_{(10)} = \frac{9}{10} \cdot \sum_{k=0}^{\infty} \left(\frac{1}{10}\right)^k = \frac{9}{10} \cdot \frac{1}{1 - \frac{1}{10}} = 1.$$

Racionális számok karakterizációja

Egy $r \in \mathbb{R}$ szám pontosan akkor racionális, ha tetszőleges b alapú számrendszerben véges, vagy végtelen szakaszos b -adikus tört.

Véges b -adikus racionális számok karakterizációja

Legyen $q = \frac{r}{s} \in \mathbb{Q}$, ahol $r \in \mathbb{Z}$, $s \in \mathbb{Z}_+$ és $\text{lnko}(r, s) = 1$, azaz az $\frac{r}{s}$ egy redukált tört. A q szám pontosan akkor véges b -adikus tört, ha $s = 1$, vagy s prímtényezős előállításában csak olyan prímek szerepelnek, amelyek a b szám prímtényezős előállításában is szerepelnek.

Példa

Könnyű látni, hogy az $\frac{1}{2}$ a tízes számrendszerben illetve a kettes számrendszerben véges, de a hármas számrendszerben végtelen szakaszos tört.

$$\frac{1}{2} = 0.5_{(10)},$$

$$\frac{1}{2} = 0.1_{(2)},$$

$$\frac{1}{2} = 0.1111\dots = 0.\dot{1}_{(3)}.$$

Az utolsó egyenlőség könnyen kijön a mértani sor összegképlete alapján:

$$0.\dot{1}_{(3)} = \sum_{k=1}^{\infty} \left(\frac{1}{3}\right)^k = \frac{1}{3} \sum_{k=0}^{\infty} \left(\frac{1}{3}\right)^k = \frac{1}{3} \cdot \frac{1}{1 - \frac{1}{3}} = \frac{1}{3} \cdot \frac{3}{2} = \frac{1}{2}.$$

Pozitív egész szám felírása b alapú számrendszerbe

Egy pozitív egész számot úgy írunk fel a b alapú számrendszerben, hogy

- maradékos osztások sorozatát végezzük,
- az osztó a b szám,
- a maradékot a szám mellé,
- a hányadost a szám alá írjuk,
- mindaddig ismételjük az eljárást, amíg a hányados nem válik nullává,
- a maradékok sorozata adja a szám jegyeit, a kiolvasás lentről felfelé történik.

Feladat

Írjuk fel a 113-at, a 2-es, a 3-as, a 4-es, a 8-as és 16-os számrendszerben. A megoldás kulcsa, hogy a 113-at fel kell írni a kettes számrendszerbe, majd mivel a $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, így a kettes számrendszerbeli jegyeket kettesével, hármásával négyesével csoportosítva megkapjuk a 113 jegyeit a 4, 8, 16 alapú számrendszerben.

Kettes számrendszerben: ($b = 2$)

113		1
56		0
28		0
14		0
7		1
3		1
1		1
0		

Így kapjuk, hogy $113 = 1110001_{(2)}$.

Hármas számrendszerben: ($b = 3$)

113		2
37		1
12		0
4		1
1		1
0		

Tehát: $113 = 1101_{(3)}$.

A négyes számrendszerben: $(b = 4) 113 = 1|11|00|01_{(2)} = 1301_{(4)}$.

A nyolcas számrendszerben: $(b = 8) 113 = 1|110|001_{(2)} = 161_{(8)}$.

A tizenhatos számrendszerben : $(b = 16)$

$113 = 111|0001_{(2)} = 71_{(16)}$. Eredményeinket foglaljuk össze egy táblázatban:

2	1110001
3	11012
4	1301
8	161
10	113
16	71

Egy szám jegyeinek számáról különböző alapú számrendszerekben

Tétel *Egy $a \in \mathbb{Z}_+$ szám számjegyeinek száma egy b alapú számrendszerben $\lfloor \log_b(a) \rfloor + 1$.*

Példa $\lfloor \log_2(113) \rfloor + 1 = 7$

Egy 0 és 1 közé eső szám felírása b alapú számrendszerbe

Egy 0 és 1 közé eső számot úgy írunk fel a b alapú számrendszerben, hogy

- szorozzuk b -vel,
- a szorzat egész részét a szám mellé,
- a törtrészét a szám alá írjuk,
- az eljárást addig ismételjük, amíg a törtrész nem válik nullává,
- akkor is leállunk, ha a törtrészek ciklikus ismétlődését tapasztaljuk,
- az egészrészek adják a számjegyeket,
- a kiolvasás fentről lefelé történik.

Feladat

Írjuk fel 2-es számrendszerben a 0.625-öt.

Mo.:

0.625		1
0.25		0
0.5		1
0		

Tehát $0.625_{(10)} = 0.101_{(2)}$.

3. Lebegőpontos számábrázolás

IEEE 754

Az **IEEE 754** (1985) (Ejtsd: "áj tripl í") szabványt ismertetjük, amely előírja a számok lebegőpontos ábrázolásának módját. Az IEEE egy betűszó, ami az (Institute of Electrical and Electronics Engineers) szavak kezdőbetűiből áll.

Négyféle **lebegőpontos számformátum** létezik,

- az **egyszeres**,
- a **dupla**,
- a **kiterjesztett** és
- a **négyszeres** pontosságú.

Ezek egymástól csak a komponenseinek bitszámaiban térnek el, ugyanúgy működnek, ezért csak az egyszeres pontosságú számábrázolást ismertetjük.

IEEE 754 szabvány ismertetése egyszeres pontosság esetén

Egyszeres pontosság esetén a számot 32 biten (= 4 byte) ábrázoljuk. A lebegőpontos számábrázoláshoz először fel kell írni a számot 2-es számrendszerben. Ha $x \in \mathbb{R}$ ($x \neq 0$) a szám, akkor

$$x = \pm 1.f_{(2)} \cdot 2^e.$$

Ezt az alakot **normált alaknak** hívjuk.

- s : 1 bit **előjelbit**. Ha a szám pozitív, akkor $s=0$, ha negatív, akkor $s=1$. (Később látni fogjuk, hogy a 0 lehet pozitív is és negatív is.)
- k eltol kitevő 8 bit. $k = e + b$ **karaktisztika**, ahol k az **eltolt kitevő** (kettes számrendszerben), e a **valós kitevő**, valamint $b = 127$. Amennyiben az eltol kitevő nyolcnál kevesebb jegyből áll, akkor az eltol kitevő elé annyi nullát írunk, hogy összesen nyolc bitet kapjunk.

- f **mantissa**, ami a maradék 23 biten van tárolva. Mivel az x szám nem lehet 0, normált alakban a diadikus pont előtt mindig áll egy egyes, így ezt az 1-est nem tároljuk, erre az átírás és a visszaírás során egyaránt oda kell figyelni. Az f bitjeit (a 32 db. bit közül a tizediktől kezdve) beírjuk a megfelelő helyre, ha f kevesebb bitből áll, mint 23, akkor a hiányzó biteket nullákkal pótoljuk.
- A kapott bitsorozatot egy hexadecimális számmá alakítjuk. Ez a hexadecimális szám a x szám egyszeres pontosságú lebegőpontos alakja.

Feladat

Ábrázoljuk a 113.625-öt lebegőpontosan egyszeres pontossággal (4 byte=32 bit).

Mo.:

- 1 lépés: *Átírás 2-es számrendszerbe.*

$$113.625 = 1110001.101_{(2)}$$

- 2 lépés: *Átírás normált alakra.*

$$1110001,101_{(2)} = 1.110|0011|01_{(2)} \cdot 2^6$$

- 3 lépés: *Az eltolt kitevő kiszámítása. A valós kitevő: $e = 6$, az eltolt kitevő: $k = e + b = 6 + 127 = 133 = 100|0010|1_{(2)}$.*

- 4 lépés: *Elhelyezés a megfelelő helyre*

4	2	<i>E</i>	3	4	0	0	0
0100	0010	1110	0011	0100	0000	0000	0000

- 5 lépés: *visszatérés a hexadecimális számmal. A lebegőpontos alak: 42E34000.*

A következő táblázat segít az inverz feladat megoldásában.

0	0000	4	0100	8	1000	<i>C</i>	1100
1	0001	5	0101	9	1001	<i>D</i>	1101
2	0010	6	0110	<i>A</i>	1010	<i>E</i>	1110
3	0011	7	0111	<i>B</i>	1011	<i>F</i>	1111

Inverz feladat

Melyik szám lebegőpontos alakja a C28C0000.

$$\begin{array}{c|c|c|c|c|c|c|c} C & 2 & 8 & C & 0 & 0 & 0 & 0 \\ 1100 & 0010 & 1000 & 1100 & 0000 & 0000 & 0000 & 0000 \end{array}$$

$s = 1$, így a szám negatív.

$$k = 1 \overset{7)}{0000} 1 \overset{2)}{0} \overset{0)}{1}_{(2)} = 2^7 + 2^2 + 1 = 133 \text{ az eltolt kitevő}$$

$$k = k - 127 = 133 - 127 = 6 \text{ a valós kitevő}$$

$$\begin{aligned} \text{A normált alak: } & -1,00011 \cdot 2^6 = - \overset{6)}{1} \overset{2)}{000} \overset{2)}{11} 0 = \\ & -(2^6 + 2^2 + 2^1) = -(64 + 4 + 2) = -70 \end{aligned}$$

Feladat

Ábrázoljuk lebegőpontosan a 0.1-et.

0.1		0
0.2		0
0.4		0
0.8		1
0.6		1
0.2		

$$0.1 = 0.0001\overline{1}_{(2)} = 1.\overline{1001}_{((2))} \cdot 2^{-4}$$

valós kitevő: $e = -4$

eltolt kitevő: $k = e + 127 = -4 + 127 = 123 = 11|1101|1_{(2)}$

123		1
61		1
30		0
15		1
7		1
3		1
1		1
0		

$$f = \overline{1001} = 100|\overline{1100}$$

0011		1101		1100		1100		1100		1100		1100		1100
3		<i>D</i>		<i>C</i>		<i>C</i>		<i>C</i>		<i>C</i>		<i>C</i>		<i>C</i>

Túlcsordulás és alulcsordulás kétszeres pontosság esetén

	előjelbit (s) 1 bit	eltolt kitevő (k) 8 bit	mantissza (f) 23 bit
NaN nem szám	0/1	11111111	$\neq 0 \dots 0$
\pm végtelen	0/1	11111111	$0 \dots 0$
lebegőpontos szám	0/1	00000001 és 11111110 között	bármilyen
denormált szám	0/1	00000000	$\neq 0 \dots 0$
nulla	0/1	00000000	$0 \dots 0$

Különböző pontosságú számábrázolás az IEEE 754-ben

	s	k	f	Σ
short real (egyszeres)	1	8	23	32
long real (dupla)	1	11	52	64
temporary real (kiterjesztett)	1	15	64	80
quad real (négyszeres)	1	15	112	128

KÖSZÖNÖM A FIGYELMET!