

Bináris aritmetika

Bevezetés

Gottfried Wilhelm Leibniz (Lipcse, 1646. július 1. – Hannover, 1716. november 14.) polihisztor: jogász, diplomata, történész, matematikus, fizikus és filozófus egyszerre. Nagy Frigyes azt mondta róla: „önmagában egy akadémia”.

Leibniz a XVII. század vége és a XVIII. század eleje között alkotott, egyike volt a német felvilágosodás alapítóinak. Newtontól függetlenül létrehozta a matematikai analízist. Leibniz hozzájárult a formális logika megteremtéséhez, az univerzális, tudományos kalkulus bevezetésével - Descartes-hoz hasonlóan - az általános megismerési módszert kereste.

A kettes számrendszer pontos leírását is ő adta meg először az Explication de l'Arithmétique Binaire című könyvében.

TABLE 86 MEMOIRES DE L'ACADEMIE ROYALE	
DES NOMBRES.	bres entiers au-dessous du double du plus haut degré. Car icy, c'est com- me si on disoit, par exemple, que III ou 7 est la somme de quatre, de deux Et que II01 ou 13 est la somme de huit, quatre & un. Cette propriété sert aux Essayeurs pour peser toutes sortes de masses avec peu de poids, & pourroit servir dans les monnoyes pour don- ner plusieurs valeurs avec peu de pieces.
0	10000000
1	10000001
2	10000010
3	10000011
4	10000100
5	10000101
6	10000110
7	10000111
8	10001000
9	10001001
10	10001010
11	10001011
12	10001100
13	10001101
14	10001110
15	10001111
16	10010000
17	10010001
18	10010010
19	10010011
20	10010100
21	10010101
22	10010110
23	10010111
24	10011000
25	10011001
26	10011010
27	10011011
28	10011100
29	10011101
30	10011110
31	10011111
32	10100000
&c.	&c.

Car icy, c'est com-
me si on disoit, par exemple, que III
ou 7 est la somme de quatre, de deux
Et que II01 ou 13 est la somme de huit, quatre
& un. Cette propriété sert aux Essayeurs pour
peser toutes sortes de masses avec peu de poids,
& pourroit servir dans les monnoyes pour don-
ner plusieurs valeurs avec peu de pieces.

Cette expression des Nombres étant établie, sert à
faire tres-facilement toutes sortes d'operations.

Pour l'Addition
par exemple.

$$\begin{array}{r} 110 \parallel 6 \\ 111 \parallel 7 \\ \hline 1101 \parallel 13 \end{array} \quad \begin{array}{r} 101 \parallel 5 \\ 1011 \parallel 11 \\ \hline 10000 \parallel 16 \end{array} \quad \begin{array}{r} 1110 \parallel 14 \\ 10001 \parallel 17 \\ \hline 11111 \parallel 31 \end{array}$$

Pour la Sou-
straction.

$$\begin{array}{r} 1101 \parallel 13 \\ 111 \parallel 7 \\ \hline 110 \parallel 6 \end{array} \quad \begin{array}{r} 10000 \parallel 16 \\ 1011 \parallel 11 \\ \hline 101 \parallel 5 \end{array} \quad \begin{array}{r} 11111 \parallel 31 \\ 10001 \parallel 17 \\ \hline 1110 \parallel 14 \end{array}$$

Pour la Mul-
tiplication.

$$\begin{array}{r} 11 \parallel 3 \\ 11 \parallel 3 \\ \hline 11 \parallel 9 \end{array} \quad \begin{array}{r} 101 \parallel 5 \\ 11 \parallel 3 \\ \hline 101 \parallel 15 \end{array} \quad \begin{array}{r} 101 \parallel 5 \\ 101 \parallel 5 \\ \hline 1010 \parallel 25 \end{array}$$

Pour la Division.

$$\begin{array}{r} 15 \parallel 1111 \\ 3 \parallel 1111 \\ \hline 5 \end{array}$$

Et toutes ces operations sont si aisées, qu'on n'a jamais
besoin de rien essayer ni deviner, comme il faut faire
dans la division ordinaire. On n'a point besoin non-plus
de rien apprendre par cœur icy, comme il faut faire dans
le calcul ordinaire, où il faut sçavoir, par exemple, que
6 & 7 pris ensemble font 13; & que 5 multiplié par 3
donne 15, suivant la Table d'une fois un est un, qu'on ap-
pelle Pythagorique. Mais icy tout cela se trouve & se
prouve de source, comme l'on voit dans les exemples pré-
cedens sous les signes $\odot$ & $\ominus$.

Az alapvető matematikai műveleteket bináris számokkal is elvégezhetjük. A bináris elven működő számítógépek is ilyen módon működnek.

A digitális számítógépekben az Aritmetikai és Logikai Egység (Arithmetical Logical Unit, ALU) végzi ezeket a műveleteket.

Ebben a leckében áttekintjük és példával megmutatjuk a:

- bináris összeadást,
- bináris kivonást,
- bináris szorzást.

Bináris összeadás

Ha két egybites számot összeadunk, a kimenetel négyféle lehet:

$\backslash (0 + 0 = 0 \backslash 1 + 0 = 1 \backslash 0 + 1 = 1 \backslash 1 + 1 = 10 \backslash)$

Az összeadást ugyanúgy végezzük, mint tízes számrendszerben. Az egyes helyi-értéktől kezdve összeadjuk a biteket és folytatjuk a magasabb helyi-értékek felé.

Látható, hogy az eredmény nem minden esetben fér el egy biten. Amikor az összeadáskor olyan eredményt kapunk, amelyiknél átvitel van, akkor egy bit túlcserdül a magasabb helyi-érték irányában. Tulajdonképpen a fenti táblázatot ki kell egészítenünk úgy, hogy a két bit összeadásakor a túlcserdülés bitet is figyelembe vesszük.

$\backslash (0 + 0 + 0 = 0 \backslash 0 + 0 + 1 = 1 \backslash 0 + 1 + 0 = 1 \backslash 0 + 1 + 1 = 10 \backslash 1 + 1 + 1 = 11 \backslash)$

Példa

```
1001101
+0010010
-----
1011111
```

Az összeadást ugyanúgy végezzük, mint tízes számrendszerben. Az egyes helyi-értéktől kezdve összeadjuk a biteket és folytatjuk a magasabb helyi-értékek felé.

Példa átvitelbittel 1.

```
  11  1  <- Átvitelbitek
1001001
+ 0011001
-----
1100010
```

Ebben a példában már vannak átvitt bitek is, ezeket a legfelső sorban jelöltük kék színnel. Az összeadás itt is ugyanúgy zajlik, mint a tízes számrendszerben.

Példa átvitelbittel 2.

```
    11    <- Átvitelbitek
  1000111
+ 0010110
-----
  1011101
```

Ebben a példában is vannak átvitt bitek. Figyeljük meg, hogy van olyan szituáció, amikor az átvitelbit további helyi-értékek felé csúszik el.

Bináris kivonás

Negatív számok esetén a komplement alapú számábrázolás az egyik lehetséges számábrázolási forma. Egy bináris szám egyes komplementét úgy kapjuk meg, hogy megcseréljük a biteket: 0-ból 1 lesz, 1-ből 0.

Például a 1010 0011 bitsorozat egyes komplemente 0101 1100 lesz.

Azért ennek az ábrázolásnak vannak hátrányai. Például a nullát a 0000 0000 és az 1111 1111 egyaránt reprezentálja.

Egy bináris szám kettes komplementét úgy képezzük, hogy az egyes komplementhez hozzáadunk egyet.

Tehát legyen az eredeti szám 1010 0011.

Egyes komplemente 0101 1100.

Kettes komplemente: $01011100 + 1 = 01011101$.

A kivonás helyett a kivonandó kettes komplementét adjuk hozzá a kisebbítendőhöz. Vegyünk egy példát:

$\backslash (7_{(10)} - 5_{(10)}) \backslash$

Binárisan ez így nézne ki: $\backslash (0111_{(2)} - 0101_{(2)}) \backslash$

Az 5 egyes komplemente 1010, kettes komplemente 1011 A kivonást tehát felcseréljük a kivonandó kettes komplementének hozzáadásához

```
  0111
+1011
-----
 10010
```

Az eredményből egyszerűen elhagyjuk a legnagyobb helyi-értékű bitet. Az eredmény tehát:

$\backslash (0111_{(2)} - 0101_{(2)}) = 0010_{(2)} \backslash$

$$\setminus (7_{(10)} - 5_{(10)}) = 2_{(10)} \setminus$$

Bináris szorzás

```
0101 * 0111
-----
0000
0101
0101
0101
-----
100011
```

Tíz-es számrendszerben ez 35-öt ad - ahogyan vártuk.

From:
<https://edu.iit.uni-miskolc.hu/> - Institute of Information Science - University of Miskolc

Permanent link:
https://edu.iit.uni-miskolc.hu/tanszek:oktatas:infrendalapjai_architekturak:logika_alapjai:binaris_aritmetika

Last update: **2024/11/11 19:37**

