



正数和負数

解放军战士社

正 数 和 負 数

張 立 編 写

解 放 軍 战 士 社

正数和負

吳立編寫

解放軍戰士社出版

解放軍印刷廠印刷

1960年6月出版 冊号 60—14—157

目 录

一、負数是什么样的数..... (1)

不够减怎样办

意义相反的数量

数轴

有“数字”愈大反而愈小的“数”吗

二、正、負数的計算..... (11)

加法

减法

乘法

除法

三、正、負数的計算規律..... (22)

加法交換律

加法結合律

减法可以改作加法算

乘法交換律

乘法結合律

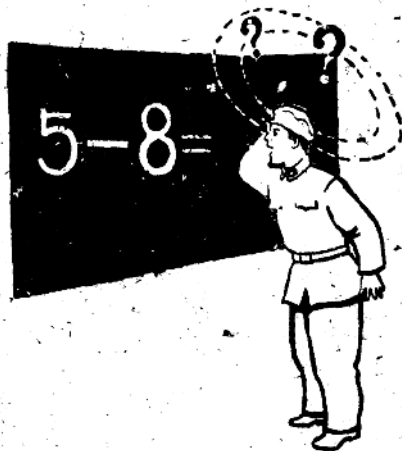
乘法分配律

一、負數是什麼樣的數

不夠減怎麼辦

$5 - 8 = ?$ 這道題在算術里是沒有辦法計算的。的確，我們在算術里所計算的減法題，都是從一個比較大的數中減去一個比較小的數；或者從一個數中減去和它相等的數，從來也沒有從一個比較小的數中減去一個比較大的數的題。看來我們好象走進了死胡同。

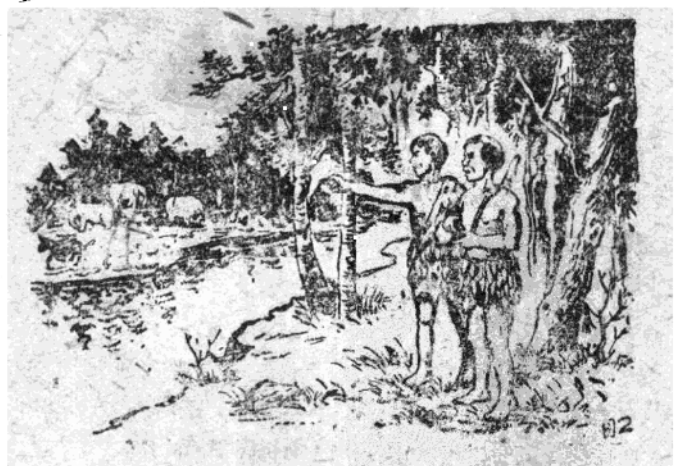
數學這條路從來就是這樣：當人們走到死胡同的時候，就要把路上的障礙拆掉，繼續前進。原始人類，對於“數”的概念只有一個、兩個、三個……等等就足夠用了。因為那時候，他們的敵人是野獸，見了野獸，大致看一看有多少，告訴他的同伴來多少人捕打就行了，用不着很詳細地計算。



后来，人們要进行交换，光会数一、数二，就没办法計算交换的东西。于是，人們又漸漸扩充了数的范围。开始，人們可以数到一百，后来又数到一千、一万。計算方法不光有加减，还有乘除。在算术里，加法、乘法、除法都能“通行无阻”，可是，碰到减法，却有一个条件：一定要被减数大于减数。如果减数大于被减数时，不够减，就没办法了。

我們怎样走出这条死胡同呢？

我們回忆一下，学习算术的时候，总是先学习整数，再学习分数。为什么学了整数还要学习分数呢？这是因为整数在計数中不够用。例如三个人分一丈布，每人分得3尺3寸3分3厘3……。而1被3除永远除不尽。写成的式子是“ $1 \div 3 = 0.333333\cdots$ ”。为了解决除不尽的問題，就需要产生一种新的数，这种数叫做分数。“ $1 \div 3$ ”就可以写做



“ $\frac{1}{2}$ ”。“ $\frac{1}{2}$ ”就是一个分数。这样，任何两个数相除（当然，零不做除数）总有一个商数了。

现在为了解决不够减的问题，又要产生一种什么样的数呢？我们且先来分析一下，够减与不够减的题。

例如： $5 - 2$ 剩余 3， 写做 $5 - 2 = 3$ ；

$5 - 3$ 剩余 2， 写做 $5 - 3 = 2$ ；

$5 - 4$ 剩余 1， 写做 $5 - 4 = 1$ ；

$5 - 5$ 没有剩余，写做 $5 - 5 = 0$ ；

$5 - 6$ 不够 1；

$5 - 7$ 不够 2；

$5 - 8$ 不够 3；

“不够”和“剩余”的意义是相反的。我们用 $+1$ 、 $+2$ 、 $+3$ 、……来表示剩余 1、剩余 2、剩余 3、……。用 -1 、 -2 、 -3 、……来表示不够 1、不够 2、不够 3、……。

于是： $5 - 2 = +3$ ；

$5 - 3 = +2$ ；

$5 - 4 = +1$ ；

$5 - 5 = 0$ ；

$5 - 6 = -1$ ；

$5 - 7 = -2$ ；

$5 - 8 = -3$ ；

-1 、 -2 、 -3 、……这种数我们把它叫做“负数”；同时我们把 $+1$ 、 $+2$ 、 $+3$ 、……这种数叫做“正数”。

负数前面的符号“ $-$ ”叫做负号；正数前面的符号“ $+$ ”

叫做正号。“-3”讀做“負3”；“+3”讀做“正3”。

“+”“-”当作正負号时，叫做性質符号；“+”“-”，当作加減号时，叫做运算符号。

沒有正号或負号的数，就是正数；所以算术中的“数”除零以外都是正数。例如3， $\frac{1}{2}$ ，0.7都是正数。因此正数前面的正号“+”可以省掉不写。

运用“負数”，就可以計算从一个比較小的数中减去一个比較大的数的題，解决不够減的問題，迈开步伐走出这条算术里的死胡同。这样，任何两个数相減也都有一个差数了。

意义相反的数量

前节講了，“不够1”和“剩余1”是两个相反意义的数量。用“-”表示“不够”，“+”表示“剩余”。为什么用“-”和“+”符号来表示相反的意义呢？

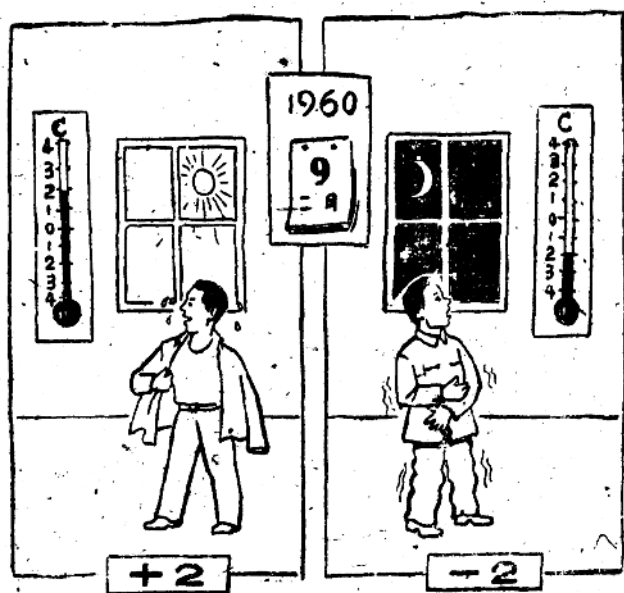
傳說在五百多年前，当时科学还不發达，人們計数的方法也很簡單。一些卖酒的人，为了計算买卖的好坏，使用横綫“—”記酒桶里的酒卖出了多少，当再把新酒灌入酒桶时，就将綫条勾銷，在原来那条横的短綫条上豎着画一道，就成了“+”，灌回了多少酒就这样勾銷多少条。最后，看所画的符号，便知道卖出了多少酒，又灌回了多少酒。

“卖出多少酒”和“灌回多少酒”是相反意义的两个数量，“-”和“+”也就表示了两种相反的意义。随后，人們逐渐便把这种簡單明确的办法，用来記載生活中許多具有

相反意义的数量，并逐步运用到了数学上去。

例如：中午的温度是零上2度，就写做+2度；午夜的温度是零下2度，就写做-2度。又如：收入5元写做+5元；支出5元写做-5元。“零上”和“零下”、“收入”和“支出”都是相反的意义。所以，凡是具有相反意义的两种数量，都可以用“正数”和“负数”分别来表示。

具有相反意义的数量，在日常生活当中，是很多的，



例如：

和前进100米意义相反的数量是后退100米；

和向东5里意义相反的数量是向西5里

和高15米意义相反的数量是低15米；

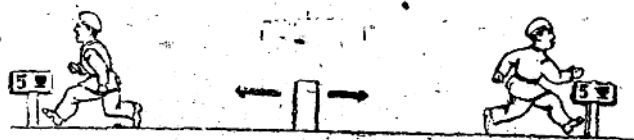
和上升9度意义相反的数量是下降9度；

和节约200元意义相反的数量是浪费200元；

和存入银行20元意义相反的数量是向银行提取20元；

和增加80人意义相反的数量是减少80人；

和10年后意义相反的数量是10年前。



在这些具有相反意义的数量里，我們規定一个为正的，規定另一个为負的。到底取哪一个数量为正，就要看我們的習慣。但是既然确定了一个数量为“正”，和它意义相反的数量就得为“負”了。

又如：在軍事學習中常用到正数和負数，例如：在炮兵射击中，向右修正的密位数为正数，向左修正的密位数为負数。

在炮兵射击中，远彈远于目标的米数为正数，近彈近于目标的米数为負数。



在地形学中，坐标在右边为正数，在左边为负数；在上方为正数，在下方为负数。

在工兵筑城中，积土高的米数为正数，挖土深的米数为负数。

在教学上为了把两种意义相反的数量统一起来说，亏损50元不说亏损，而说盈余-50元（把亏损和盈余都说成盈余）；不说下降5米，而说上升-5米；（把下降和上升都说成上升）不说5年前，而说-5年后等等。这样，如果说我身边有錢-3元，实际上我身边沒有錢而是欠人家3元錢；如果说我比他高一20厘米，实际上我不比他高，而是比他矮20厘米。

正数和负数既然是表示意义相反的两类量的数；于是我们就把-3叫做+3的相反数；同样+3也是-3的相反数。

如果我们不管原来的数的性质，只看它的“数字”，那么，就叫这个数的绝对值。例如我们有时说甲住在北京的东面100里，乙住在北京的西面100里，这就考虑了方向（也就是性质）；我们有时说甲离开北京100里，乙离开北京也是100里，这就是按绝对值来说的。所以，“-”和“+”当作正负号时，是表示了数量的性质，叫做性质符号。

我們把某數的前后各加一豎線來表示這數的絕對值。例如 -3 的絕對值是 3 寫做 $|-3| = 3$ ； $+3$ 的絕對值是 3 寫做 $|+3| = 3$ 。同樣的 $|-1\frac{1}{2}| = 1\frac{1}{2}$ ； $|+2\frac{1}{2}| = 2\frac{1}{2}$ 等等。換句話說，正數的絕對值就等於它自己；而負數的絕對值就是去掉負號后所得到的正數。

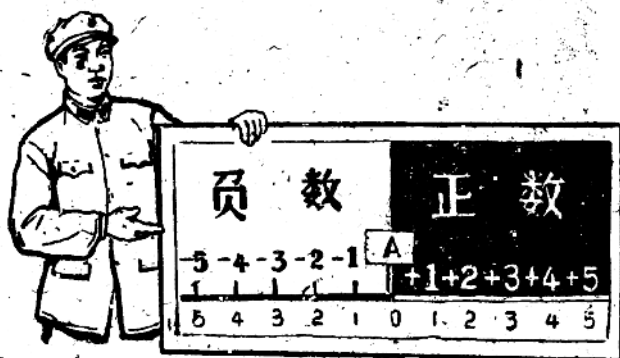
數 軸

機器上有“軸”，車輪上有“軸”，難道數量也有“軸”嗎？有。

我們測量各種不同的量時，往往把各種量的大小化成長度來計算的。例如在溫度計上刻一條線來計量溫度的高低，線上一固定點為零度，向上刻有 1 、 2 、 3 、……等分劃，表示零上 1 度、零上 2 度、零上 3 度、……等等；向下也刻有 1 、 2 、 3 、……等分劃表示零下 1 度、零下 2 度、零下 3 度、……等等。

為了表示正數、負數與零之間的關係和研究數量的變化，在教學上用直線上的點表示數。怎樣表示法呢？我們任意畫一條直線，在這條直線上任意取一個點（例如 A 點）為起點（數學上叫做原點），我們規定這條直線從原點起由左到右的方向是正方向（我們畫線的時候就是從左往右畫的）；和它相反的方向就是負方向，也就是說從右到左的方向是負方向。我們還要取一個適當的長度，做測量的單位。用所取的長度單位從 A 點起往右邊測量，把原點 A 作“ 0 ”就得到 $+1$ 、 $+2$ 、 $+3$ 、 $+4$ 、 $+5$ 、……等等的點。用所取的

長度單位從A點往左邊測量就得到 -1 、 -2 、 -3 、 -4 、 -5 、……等等的點。不僅可以得到整數的點，還可以得到分數的點。例如 $+\frac{1}{2}$ 、 $+\frac{2}{3}$ ……， $-\frac{1}{2}$ 、 $-\frac{2}{3}$ ……等等。這樣每一個數都可以用這條直線上的點來表示。這條用來表示數的直線叫做數軸。



從數軸上我們可以看出，如果在原點的右邊有一點表示一個正數（例如 $+3$ ），那末在原點左邊也必定有一個離開原點同樣距離的點表示一個負數（例如 -3 ），這個負數就是那個正數的相反數。反過來說，如果在原點的左邊有一點表示一個負數（例如 -5 ），那末在原點的右邊也必定有一個離開原點同樣距離的點表示一個正數（例如 $+5$ ），這個正數就是那個負數的相反數。

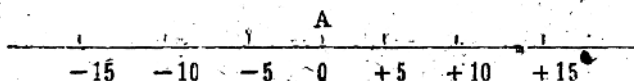
有“数字”愈大反而愈小的“数”吗

“5”和“3”誰大，当然是“5”大、“3”小，写做 $5 > 3$ ，这是正数的比大小；要問“-5”和“-3”誰大，那就是“-3”大，“-5”小，写做 $-3 > -5$ ，为什么数字大的反而愈小呢？这是負数的比大小。負数比大小，猛一看，似乎是錯的，其实是对的。我們只要看看下面的一些例子，就明白了。

先用温度的高低來說，-3度表示零下3度；-5度表示零下5度；零下3度比零下5度温度高些，所以-3度大于-5度。

再用經濟状况來說，-3元表示欠人家3元；-5元表示欠人家5元；欠人家錢較少的比欠人家錢較多的經濟情况要好些，因此我們說-3元大于-5元。

对于正負数的大小，我們可以用数軸来作如下的規定：凡是在数軸右边的数比在左边的数大；愈靠右的愈大，反过來說，在数軸左边的数比在右边的数小，愈靠左的愈小。



从上面講的，关于正負数的大小，我們可以得出以下几条規律：

(1) 任何一个正数都大于零。例如：

$$+5 > 0; +2\frac{1}{2} > 0。$$

(2) 零大于任何一个负数。例如：

$$0 > -1; 0 > -4\frac{8}{9}。$$

(3) 任何一个正数都大于任何一个负数。例如：

$$+1 > -1; +3.8 > -4.9。$$

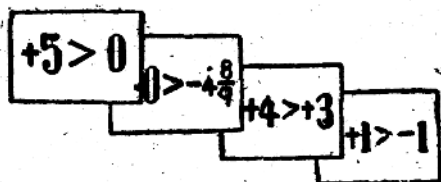
(4) 两个正数相比较，绝对值大的，正数也大。例如：

$$+4 > +3; +3.1 > +2.9。$$

(5) 两个负数相比较，绝对值大的负数反而小。例如：

$$-4 > -5; -0.9 > -1.1。$$

所以，负数的大小，恰好与正数相反，它的“数字”愈大，所表示的“数值”反而愈小。



二、正、负数的计算

正数的计算，在算术里，我们学习了，知道它是怎样计算的，也知道它在生活实际中的应用。现在，知道了什么是负数。那么，负数的计算是怎样的呢？负数和正数混合计算又是怎样的呢？它们在生活实际又有些什么用呢？在下面就来谈谈这几个问题。

加 法

在炮兵射击中，有时需要进行方向修正。如果第一次向右修正 5 密位，第二次向右修正 3 密位，那末两次一共向右修正了 8 密位。我們規定向右修正的密位数为正数，这个題列成計算的式子便是： $(+5) + (+3) = +8$

同样举炮兵射击的例子可以解釋負数加負数和正数加負数的問題。

例如第一次向左修正 5 密位，第二次又向左修正 3 密位，两次一共向左修正 8 密位。向左修正的密位数为負数，列成計算的式子是： $(-5) + (-3) = -8$

再如第一次向右修正 5 密位，第二次又向左修正 3 密位，两次实际上等于向右修正 2 密位。計算式子是：

$$(+5) + (-3) = +2$$

可能有的同志会怀疑：为什么正数与負数相加的和数比被加数小了呢？这样看是用算术的观点来看代数的計算。

因为正負数是表示意义相反两种量的数，意义相反两种数合在一起要互相抵銷了一部分，和数就要比其中的正数小些，就要用减法来計算抵銷的一部分了。

再举些例子來說：收入 5 元又支出 3 元等于只收入 2 元；前进 5 里又后退 3 里等于只前进 2 里；上升 5 度又下降 3 度等于只上升 2 度。这些例子都說明了正数和負数加到一起，有时“和数”比被加数要小些。

一种新的数的計算，有新的結果，是不奇怪的；正如我

們在計算小數和分數的乘法時，有時會發現乘得的積數比被乘數小；在計算小數和分數的除法時，有時會發現除得的商數比被除數大，這種結果是整數計算中所沒有的。

上面的正數加負數的例子中，正數的絕對值比負數的絕對值大些，加得的和數是正數；如果正數的絕對值比負數的絕對值小時，加得的和數就是負數了。

例如第一次向右修正 3 密位，第二次向左修正 5 密位，兩次實際上等於向左修正 2 密位，例成式子是： $(+3) + (-5) = -2$

如果是絕對值相等而性質符號相反的两个數（或者說兩個相反數）相加時所得和數便是零。例如第一次向右修正 5 密位，第二次向左修正 5 密位，結果等於沒有修正。再如收入 5 元又支出 5 元收支相抵銷；前進 5 里又后退 5 里仍回到原來地點；上升 5 度又下降 5 度，仍是原來的溫度。所以 $(+5) + (-5) = 0$

在正負數的加法中，我們找到了以下的規律：

(1) 正數加正數，把它們的絕對值相加，結果仍得正數。

(2) 負數加負數，把它們的絕對值相加，結果仍得負數。

(3) 正數和負數相加，把它們的絕對值相減，結果的符號，用絕對值比較大的數的符號。

(4) 兩個數字相同、符號相反的數相加，它們的和數等於零。