# 计算机的运算方法-练习 1（观看 B 站或 mooc 刘宏伟视频 6.3-6.4 节）

**1、双符号位补码也叫变形补码，在不同场合有不同含义，选出正确选项：**

1）在定点整数补码加减法中，双符号位 00 代表(B)、11 代表(C)、01 和 10 代表（A）

A、溢出　　B、正数　　C、负数

2）在浮点数补码加减法中，尾数出现双符号位 00 代表(B)、11 代表(C)、01 和 10 代表（D）

A、溢出　　B、正数　　C、负数　　D、非规格化数，需要进行规格化处理

3）Booth 算法中的双符号位从左至右分别代表什么含义？（C）

A、都是符号位

B、都是数值位

C、符号位和数值部分的进位

D、数值部分的进位和符号位

**2、填表（表格需要熟练记忆）** 令n代表数值部分位数

| 算法 | 加法次数 | 移位次数 | 移位（填写算术或逻辑、左移或右移） | 符号是否参与运算 |
|---|---|---|---|---|
| 原码一位乘 | n | n | 部分积逻辑右移 | 否 |
| Booth 算法 | n+1 | n | 部分积算术右移 | 是 |
| 加减交替法，（也叫不恢复余数法） | n+1或n+2 | n | 余数逻辑左移 | 否 |

**3、计算题**（**给出计算过程详细步骤**）

**1）补码加减法类题型（重点）**

设机器数字长为 8 位（含 1 位符号位），用**补码运算规则**求 A-B。

其中 A=$-\frac{10}{64}$，B=$-\frac{21}{128}$。

**2）原码一位乘题型**

已知二进制数 x=-0.1100，y=0.1001，按**原码一位乘**计算 x＊y。

（答案：-0.01101100）

**3）Booth 算法题型（重点）**

已知二进制数 x=-0.1011，y=-0.1101，按 **Booth 算法**计算$[x*y]_{补}$及其真值。

(答案：0.10001111)

**4）原码一位除，加减交替法（即不恢复余数法）**

已知二进制数 x=-0.1001，y=0.1101，用原码加减交替法计算$[x/y]_{原}$，并给出商与余数的真值。（答案：1.1011; -0.1011; -0.0001*$2^{-4}$）

**5）浮点数加减法题型（需要掌握）**

已知：$x = 2^{-011} \times 0.101100$，$y = 2^{-010} \times (-0.011100)$，求$[x \pm y]_{补}$

**（观看 https://www.bilibili.com/video/BV1t4411e7LH?p=90 6.4 节的 3 个视频做题）**

## 1）补码加减法类题型（重点）

设机器数字长为 8 位（含 1 位符号位），用**补码运算规则**求 A-B。其中 $A=-\frac{10}{64}$，$B=-\frac{21}{128}$。

0.001010

解：$A=-\frac{10}{64} \Rightarrow [A]_{原}=0.001010$

$[A]_{补}=1.110110$

$B=-\frac{21}{128} \Rightarrow -B=\frac{21}{128}$

$-B>0 \Rightarrow [-B]_{原}=[-B]_{补}=0.0010101$

$[A-B]_{补}=[A]_{补}+[-B]_{补}$

$$
\begin{array}{r}
=1.1101100 \\
+\ 0.0010101 \\
\hline
\boxed{1}0.0000001
\end{array}
$$

丢掉

（小数末尾补0，目的是两个数对齐）

而且本题要求8位

$\therefore [A-B]_{补}=0.0000001$

$\therefore A-B=0.0000001=\frac{1}{128}$

注意：①默认用1位符号位进行加法操作，化减为加

②进位丢掉要写一下，同时溢出要标注（答题规范）

例：

$$
\begin{array}{r}
x_0.x_1x_2\cdots x_n \\
+\ y_0.y_1y_2\cdots y_n \\
\hline
z_0.z_1z_2\cdots z_n
\end{array}
$$

丢掉

如果有溢出也要写“溢出”二字。

**2）原码一位乘题型**

已知二进制数 x=-0.1100，y=0.1001，按**原码一位乘**计算 x * y。

（答案：-0.01101100）

解：第1步：写出原码、绝对值和符号位

$[x]_{原}=1.1100$，$x^*=0.1100$（绝对值），$x_0=1$

$[y]_{原}=0.1001$，$y^*=0.1001$（绝对值），$y_0=0$

第2步：$x^* \times y^*$ 计算过程

| 部分积 | 乘数 | |
|---|---|---|
| 0.0000 | 1001 | |
| + 0.1100 | | $+x^*$ |
| 0.1100 | | |
| 0.0110 | 0100 | → |
| + 0.0000 | | +0 |
| 0.0110 | | |
| 0.0011 | 0010 | → |
| + 0.0000 | | +0 |
| 0.0011 | | |
| 0.0001 | 1001 | → |
| + 0.1100 | | $+x^*$ |
| 0.1101 | | |
| 0.0110 | 1100 | → |

点评：原码一位乘

根据递推公式：

n次加法

n次移位

（n是数值部分位数）

加一次移位一次

横线表示加法相加过程

第3步：写出运算结果

① 乘积符号位 $x_0 \oplus y_0 = 1$

② $x^* \cdot y^* = 0.01101100$

则 $x \cdot y = -0.01101100$

3）Booth 算法题型 **（重点）**

已知二进制数 x=-0.1011，y=-0.1101，按 **Booth 算法**计算[x * y]补及其真值。

(答案：0.10001111)

点评：要记住这个表

| $y_i$ $y_{i+1}$ | $y_{i+1}-y_i$ | 操作 |
|---|---|---|
| 0 0 | 0 | →1 |
| 0 1 | 1 | +[x]补 →1 |
| 1 0 | -1 | +[−x]补→1 |
| 1 1 | 0 | →1 |

解：第1步写出 [x]补，[-x]补，[y]补

[x]补 = 1.0101

[-x]补 = 0.1011

[y]补 = 1.0011

第2步：被乘数用双符号位

乘数用于加法规则判断

"前有符号位，后有附加位0"

竖式计算

| | | |
|---|---|---|
| 00.0000 | 1 0011 0 | |
| +00.1011 | | +[-x]补 |
| 00.1011 | | |
| 00.0101 | 1 1001 1 | → |
| 00.0010 | 1 1100 1 | → |
| +11.0101 | | +[x]补 |
| 11.0111 | | |
| 11.1011 | 1 1110 0 | → |
| 11.1101 | 1 1111 0 | → |
| +00.1011 | | +[-x]补 |
| 00.1000 | 1111 | 最后一步不移位 |

(注意：最后一列代表了本行执行了什么操作)

第3步：写出运算结果，用1位符号位表示

[x·y]补 = 0.10001111　∴ x·y = 0.10001111

### 4）原码一位除，加减交替法（即不恢复余数法）

已知二进制数 x=-0.1001，y=0.1101，用原码加减交替法计算[x / y]原，并给出商与余数的真值。（答案：1.1011; -0.1011; $-0.0001*2^{-4}$）

解：$[x]_原 = 1.1001$

$[y]_原 = 0.1101$

$[x^*]_补 = 0.1001$

$[y^*]_补 = 0.1101$

$[-y^*]_补 = 1.0011$

点评：加减交替法推导得

上商"1"　$2R_i - y^*$

上商"0"　$2R_i + y^*$

加减交替

加减交替实际是 $x^*$ 与 $y^*$ 运算，但借助了补码"化减为加"

| 被除数(余数) | 商 | 说明 |
|---|---|---|
| 0.1001 | 0.0000 | |
| + 1.0011 | | +$[-y^*]_补$ |
| 1.1100 | 0 | 余数为负，上商0 |
| 1.1000 | 0 | ← |
| + 0.1101 | | +$[y^*]_补$ |
| 0.0101 | 01 | 余数为正，上商1 |
| 0.1010 | 01 | ← |
| + 1.0011 | | +$[-y^*]_补$ |
| 1.1101 | 010 | 余数为负，上商0 |
| 1.1010 | 010 | ← |
| + 0.1101 | | +$[y^*]_补$ |
| 0.0111 | 0101 | 余数为正，上商1 |
| 0.1110 | 0101 | ← |
| + 1.0011 | | +$[-y^*]_补$ |
| 0.0001 | 01011 | 余数为正，上商1 |

必考 注意：商上满之后，余数部分小数点前为0即可停止。但当小数点前的1时，需多加一次$[y^*]_补$

(此时商不用动，因为已经算完了)

① 符号位 $x_0 \oplus y_0 = 1 \oplus 0 = 1$

② $\frac{x^*}{y^*} = 0.1011$　∴ $[x/y]_原 = 1.1011$

∴ $x/y = -0.1011$

余 $-0.0001 \times 2^{-4}$

(因为余数符号与被除数保持一致)

见PPT例题

所以可能出现n+2次加法

## 5）浮点数加减法题型（需要掌握）

已知：$x = 2^{-011} \times 0.101100$，$y = 2^{-010} \times (-0.011100)$，求$[x \pm y]_{补}$

[redacted]解：$[x]_原 = 11,011;\ 00.101100$　　$[y]_原 = 11,010;\ 11.011100$

$[x]_补 = 11,101;\ 00.101100$　　$[y]_补 = 11,110;\ 11.100100$

① 对阶，小阶向大阶对齐

$\therefore [x]'_补 = 11,110;\ 00.010110$

② $[x+y]_补 = [x]'_补 + [y]_补$，尾数求和

```
  00.010110      用[x]'补代替[x]补进行运算
+ 11.100100
-----------
  11.111010
```

$\therefore [x+y]_补 = 11,110;\ 11.111010$

③ 左规 $[x+y]_补 = 11,011;\ 11.010000$

注意：这里尾数左移3位，阶码减3，

即 $11,110 + [-3]_补 = 11,110 + 11,101 = 11,011$

④ 由于左规无精度损失，无需舍入。

$[-y]_补 = 11,110;\ 00.011100$

$[x-y]_补 = [x]'_补 + [-y]_补$，尾数求和

```
  00.010110      用[x]'补代替[x]补进行运算
+ 00.011100
-----------
  00.110010
```

$\therefore [x-y]_补 = 11,110;\ 00.110010$

由于$[x-y]_补$尾数已是规格化数，因此无需左规或右规，也无需舍入。