计算机的运算方法-练习 2

计算题（注意规范答题，并给出计算过程详细步骤）

1、补码加减法题型

1）已知 A=-1001,B=-0101,求$[A+B]_{补}$

2）设机器数字长为 8 位（含 1 位符号位），若 A=+15，B=+24，求$[A-B]_{补}$并还原成二进制真值和十进制数。

3）设机器数字长为 8 位（含 1 位符号位），若 A=-93，B=+45，求$[A+B]_{补}$

4）已知 A=-0.1000，B=-0.1000，求$[A+B]_{补}$

2、BOOTH 算法

1）已知 x= +19，y= -35，利用 BOOTH 算法求 x·y。

2）已知二进制数 x=-0.1111，y=0.1101，按 BOOTH 算法计算$[x * y]_{补}$及其真值。

(答案：1.00111101; -0.11000011)

3、浮点数加减法 （本题要求:对阶及最后的右归处理出现精度损失则采用“0 舍 1 入”法进行处理）

已知：$x = 2^{101} \times (-0.100101)$，$y = 2^{100} \times (-0.001111)$，求$[x \pm y]_{补}$。

1.1) $[A]_{补}=1,0111$

$[B]_{补}=1,1011$

$[A+B]_{补}=[A]_{补}+[B]_{补}$

$=1,0111$

$+1,1011$

$\boxed{1}1,0010$

丢掉

$\therefore [A+B]_{补}=1,0010$

2) $[A]_{补}=0,0001111$

$[-B]_{补}=1,1101000$

$[A-B]_{补}=[A]_{补}+[-B]_{补}$

$=0,0001111$

$+1,1101000$

$1,1110111$

$[A-B]_{补}=1,1110111$

$A-B=-0001001=-9$

3) [A]补 = 1,0100011

[B]补 = 0,0101101

[A+B]补 = [A]补 + [B]补

$$
\begin{array}{r}
= 1,0100011 \\
+\ 0,0101101 \\
\hline
1,1010000
\end{array}
$$

4) [A]补 = 1.1000

[B]补 = 1.1000

[A+B]补 = [A]补 + [B]补

$$
\begin{array}{r}
= 1.1000 \\
+\ 1.1000 \\
\hline
\boxed{1}\,1.0000
\end{array}
$$

去掉

[A+B]补 = 1.0000

相同符号补码情况下，
结果符号位与原操作数同号，无溢出

3延展求[A-B]补

[A]补 = 1,0100011

[-B]补 = 1,1010011

[A-B]补 = [A]补 + [-B]补

$$
\begin{array}{r}
= 1,0100011 \\
+\ 1,1010011 \\
\hline
\boxed{1}\,0,1110110
\end{array}
$$

去掉

∴ [A-B]补 = 0,1110110

由于加法的两个数符号位与结果符号位不同，因此溢出。

2、BOOTH 算法

1）已知 x= +19，y= −35，利用 BOOTH 算法求 x·y。

解：$[x]_{原}=0,010011$

$[x]_{补}=0,010011$

$[-x]_{补}=1,101101$

$[y]_{原}=1,100011$

$[y]_{补}=1,011101$

①部分积取双符号位，

②乘数左边1个符号位
右边1个附加位

部分积	乘数	附加位	
00,000000	1011101	0	
+11,101101			$+[-x]_{补}$
11,101101			
11,110110	1101110	1	→1
+00,010011			$+[x]_{补}$
00,001001			
00,000100	1110111	0	→1
+11,101101			$+[-x]_{补}$
11,110001			
11,111000	1111011	1	→1
11,111100	0111101	1	→1
11,111110	0011110	1	→1
+00,010011			$+[x]_{补}$
00,010001			
00,001000	1001111	0	→1
+11,101101			$+[-x]_{补}$
11,110101	100111		

$\therefore [x\cdot y]_{补}=1,110101100111$

$[x\cdot y]_{原}=1,001010011001$

写结果时，只写单符号即可
（双符号只用在部分积的运算中）

$\therefore x\cdot y=(-1010011001)_2$

$=(-665)_{10}$

点评：本题注意 $(19)_2=10011$，而 $(35)_2=100011$，需对19前添0，使得位数对齐。

因此有 $[x]_{原}=[+19]_{原}=0,010011$

$[y]_{原}=[-35]_{原}=1,100011$

对于整数利用Booth算法求乘法，与小数类似，区别仅在于分隔符不同，但整数要添0对齐两个数的位数。

2）已知二进制数 x=-0.1111，y=0.1101，按BOOTH 算法计算[x * y]补及其真值。

(答案：1.00111101; -0.11000011)

解：$[x]_补 = 1.0001$

$[y]_补 = 0.1101$

$[-x]_补 = 0.1111$

部分积	乘数	附加位	
00.0000	0.1101	0	$+[-x]_补$
+00.1111			
00.1111			
00.0111	10110	1	→
+11.0001			$+[x]_补$
11.1000			
11.1100	01011	0	→
+00.1111			$+[-x]_补$
00.1011			
00.0101	10101	1	→
00.0010	11010	1	→
+11.0001			$+[x]_补$
11.0011	1101		最后不移位

$[x \times y]_补 = 1.00111101$

$x \cdot y = -0.11000011$

3、浮点数加减法 本题要求对阶和会用"0舍1入"处理精度损失

已知：$x = 2^{101} \times (-0.100101)$，$y = 2^{100} \times (-0.001111)$，求$[x \pm y]_{补}$。

1、求$[x+y]_补$

① 对阶，$[y]_补' = 00,101;11.111001$（对于精度损失采取舍入处理，则：）
→ 进行了舍入处理（0舍1入，或恒置1法）

② 尾数求和

$$\begin{array}{r} 11.011011 \\ +\ 11.111001 \\ \hline 11.010100 \end{array}$$

∴ $[x+y]_补 = 00,101;11.010100$ ③ 无需规格化与舍入

2、求$[x-y]_补$

根据题目写出 $[-y]_补 = 00,100;00.001111$

① 对阶：$[-y]_补' = 00,101;00.001000$（0舍1入获得）

② 尾数求和：$[x-y]_补 = 00,101;11.100011$

③ 左规：$[x-y]_补 = 00,100;11.000110$

总结：浮点数加减法分三步：

① 对阶 ② 尾数求和 ③ 左规或右规

总结：对阶过程及右规都可能
造成精度损失
此种情况看题中要求用
"0舍1入"法还是恒置1"法。
或者也有对阶过程不做处理，直接丢掉末尾数值。
如果考试此题会明确要求具体采用何种舍入处理方法。

左规针对 { ① $11.1xxx \rightarrow 11.0xxx$
② $00.0xxx \rightarrow 00.1xxx$
连同阶码一起调整

右规针对 { ① $10.xxx \rightarrow 11.0xxx$
② $01.xxx \rightarrow 00.1xxx$

由于双符号位补码运算，00.表示正数，11.为负数，
左规和右规必须满足符号位和数值第1位
不同，连同阶码一起调整为 $11.0xxx$ 或 $00.1xxx$