



知识点总结



- 原码、反码和补码是二进制整数的不同表示方式，主要用于二进制数的加减运算。
- **原码**：最高位（即最左位）是符号位，0表示正数，1表示负数；其余位表示数值大小。
- **反码**：
 - 正数的反码：与原码相同。
 - 负数的反码：在原码的基础上，符号位不变，对其余位取反。（取反就是把1变成0，0变成1）
- **补码**：
 - 正数的补码：与原码相同。
 - 负数的补码：在反码的基础上，再加1

例如：用8位二进制表示+5、-5的原码、反码、补码。

+5的原码：00000101	-5的原码：10000101
+5的反码：00000101	↓（符号位不变，其余为取反）
+5的补码：00000101	-5的反码：11111010
	↓（加1）
	-5的补码：11111011

- 在原码和反码中，0有+0和-0两种表示形式；补码中只有一个无符号0。



· 二进制数的加法运算：

列出竖式，逢2进1。若计算结果出现溢出，则从右往左保留有效位数。比如，在8位二进制加法中，如果结果超过8位，就从右往左保留8位，舍弃溢出的部分。

例子：

1、用8位二进制补码计算 $5+(-5)$ 。

```
00000101
+ 11111011
-----
100000000
```

结果是100000000，**超过8位，则从右往左保留8位**，即00000000。

结果是补码，转为原码表示。其最高位是0，即为正数。正数的原码和补码一致，即00000000。

该原码对应的十进制是0，计算正确。



2、用8位二进制补码计算 $-3+(-8)$ 。

```
  11111101
+ 11111000
-----
```

111110101

结果是补码，111110101超过8位，从右往左保留8位，即为11110101。

将结果转为用原码表示。其最高位是1，即为负数。

负数的补码转为原码：

先减1，得到反码：11110100。

再对反码取反，得到原码：10001011。

该原码对应的十进制就是-11，计算正确。



负数的补码转原码，要先减1得到反码



题目1、已知一个数的补码为 10011010，其原码是（ A ）

A. 11100110

B. 01100110

C. 10011001

D. 10011010



负数的反码是在原码的基础上，符号位不变，其余位取反

题目2、若一个数的原码为 10010110，其反码为 11101001，补码为 11101010。



题目3、一个数的反码为 10101100, 这个数是 -83。(十进制)



题目4、计算13-7，采用 8 位二进制补码运算。

13的补码：00001101

-7的原码：10000111 -7的反码：11111000 -7的补码：11111001

00001101

+ 11111001

100000110

00000110



题目5、计算 $-9+5$ ，采用 8 位二进制补码运算。

-9的原码：10001001 -9的反码：11110110 -9的补码：11110111

5的补码：00000101

```
  11110111
+ 00000101
-----
```

11111100

-1

反码：11111011

符号位不变其余取反：10000100

十进制：-4