

# 进制转换计算方法

作者 彭一航

开始日期 2017-12-21

## 0 主旨

讲解把十进制整数或小数转换成其它进制数的计算过程，实现一个做进制转换的程序，这个程序将用 c 语言，python,shell,matlab 分别实现一遍。

## 1 十进制整数转其它进制计算方法

### 1.1 通用逻辑

用 dec 表示原十进制数，用 oba 表示目标进制，

第一步，

$$dec \div oba = res_1 \cdots rem_1$$

res 表示相除结果，rem 表示余数，

第二步，

令十进制数等于上一步相除结果，用新的十进制数继续除以目标进制，行到新的相除结果和余数，

$$dec_2 = res_1$$

$$dec_2 \div oba = res_2 \cdots rem_2$$

第三步，重复执行第二步

$$dec_3 = res_2$$

$$dec_3 \div oba = res_3 \cdots rem_3$$

$$dec_4 = res_3$$

$$dec_4 \div oba = res_4 \cdots rem_4$$

... ..

$$dec_n \div oba = 0 \cdots rem_n$$

直到整除结果为 0。

第四步，把上面每一步得到的余数，按后算出来的，或者说下面的，排在左边；先算出来的，或者说上面的，排在右边，

$$rem_n \ rem_{n-1} \ rem_{n-2} \cdots rem_2 \ rem_1$$

就是目标进制数。

### 1.2 示例

#### 1.2.1 例一 把十进制数 231 转成二进制

$$231 \div 2 = 115 \dots 1$$

$$115 \div 2 = 57 \dots 1$$

$$57 \div 2 = 28 \dots 1$$

$$28 \div 2 = 14 \dots 0$$

$$14 \div 2 = 7 \dots 0$$

$$7 \div 2 = 3 \dots 1$$

$$3 \div 2 = 1 \dots 1$$

$$1 \div 2 = 0 \dots 1$$

上面从原始的 231 开始，每次除以 2，直至结果为 0，即最后一步的  $1 \div 2 = 0$  为止。除了 8 次，得到 8 次余数，把上述余数们横向排列，下面的排在左边，上面的排在右边，

11100111

就是十进制数 231 对应的二进制。

验证：

$$2^7 + 2^6 + 2^5 + 2^2 + 2^1 + 2^0 = 128 + 64 + 32 + 4 + 2 + 1 = 231$$

### 1.2.2 例二 把十进制数 114 转成八进制

$$114 \div 8 = 14 \dots 2$$

$$14 \div 8 = 1 \dots 6$$

$$1 \div 8 = 0 \dots 1$$

把余数们按下面的排在左边，上面的排在右边的顺序排列起来得 162，

验证：

$$1 \times 8^2 + 6 \times 8^1 + 2 \times 8^0 = 64 + 48 + 2 = 114$$

### 1.2.3 例三 把十进制数 981 转成十六进制

$$981 \div 16 = 61 \dots 5$$

$$61 \div 16 = 3 \dots D$$

$$3 \div 16 = 0 \dots 3$$

把余数们按下面的排在左边，上面的排在右边的顺序排列起来，得到 3D5，

验证：

$$3 \times 16^2 + D \times 16^1 + 5 \times 16^0 = 3 \times 16^2 + 13 \times 16^1 + 5 \times 16^0 = 981$$

## 2 十进制小数转其它进制计算方法

### 2.1 通用逻辑

为简化问题，假设小数的整数部分都是 0。

用 dec 表示原十进制小数，oba 表示目标进制，

第一步，用原十进制小数乘以目标进制

$$dec \times oba = int_1 . dec_1$$

$int_1$  表示相乘结果的整数部分， $dec_1$  表示新小数部分

第二步，用第一步所得新小数乘以目标进制

$$dec_1 \times oba = int_2 . dec_2$$

第三步，重复第二步，

$$dec_2 \times oba = int_3 . dec_3$$

... ..

$$dec_{n-1} \times oba = int_n . 0$$

直到相乘结果的小数部分为零。

第四步，把上面得到的各次相乘结果的整数部分，按先算的，或者说上面的，左边；后算的，或者说下面的，排在右边，

$$int_1 \ int_2 \ \cdots \ int_n$$

就是目标进制数。

## 2.2 示例

### 2.2.1 例四 十进制小数 0.333 转二进制

	Integer Part	Decimal Part
$0.333 \times 2$	0	0.666
$0.666 \times 2$	1	0.332
$0.332 \times 2$	0	0.664
$0.664 \times 2$	1	0.328
$0.328 \times 2$	0	0.656
$0.656 \times 2$	1	0.312
$0.312 \times 2$	0	0.628
...	...	...

计算步骤是从原始数 0.333 开始，乘以 2，等于 0.666，记下 0.666 的整数部分是 0，小数部分是 0.666，

再用 0.666 乘以 2，等于 1.332，记下 1.332 的整数部分是 1，小数部分是 0.332，

反复用上一步所得小数部分乘以 2，直至相乘结果的小数部分为 0。

这种步骤可能小数部分永远不为 0，这时可以在进行了适当多步后，得到了适当精度后，放弃后面的计算。

上述只计算了 7 步，把每一步所得的整数部分，按上面的在左边，下面的在右边的顺序，横向排列起来，和十进制整数转二进制的这一步顺序相反，得到 0.010101，

验证：

$$0 \times \frac{1}{2^1} + 1 \times \frac{1}{2^2} + 0 \times \frac{1}{2^3} + 1 \times \frac{1}{2^4} + 0 \times \frac{1}{2^5} + 1 \times \frac{1}{2^6} =$$

$$1 \times \frac{1}{2^2} + 1 \times \frac{1}{2^4} + 1 \times \frac{1}{2^6} =$$

$$0.25 + 0.0625 + 0.015625 = 0.328125 \approx 0.333$$

### 2.2.2 例五 十进制小数 0.33 转成八进制

	Integer Part	Decimal Part
$0.33 \times 8$	2	0.64
$0.64 \times 8$	5	0.12
$0.12 \times 8$	0	0.96
$0.96 \times 8$	7	0.68
$0.68 \times 8$	5	0.44
$0.44 \times 8$	3	0.52
$0.52 \times 8$	4	0.16
$0.16 \times 8$	1	0.28

目标是不断乘以 8 直至小数部分为 0，这个目测小数部分永远不会为 0，只计算了 8 步，把每步所得整数部分按上面在左边，下面在右边的顺序排列起来得 0.25075341，验证

$$2 \times \frac{1}{8} + 5 \times \frac{1}{8^2} + 0 \times \frac{1}{8^3} + 7 \times \frac{1}{8^4} + 5 \times \frac{1}{8^5} + 3 \times \frac{1}{8^6} + 4 \times \frac{1}{8^7} + 1 \times \frac{1}{8^8} = 0.32999995 \approx 0.33$$

### 2.2.3 例六 十进制小数 0.33 转成十六进制

	Integer Part	Decimal Part
$0.33 \times 16 = 5.28$	5	0.28
$0.28 \times 16 = 4.48$	4	0.48
$0.48 \times 16 = 7.68$	7	0.68
$0.68 \times 16 = A.88$	A	0.88
$0.88 \times 16 = E.08$	E	0.08
$0.08 \times 16 = 1.28$	1	0.28

目标是不断乘以 16 直至小数部分为 0，这个目测小数部分永远不会为 0，只计算了 6 步，把每步所得整数部分按上面在左边，下面在右边的顺序排列起来得到 0.547AE1，验证

$$5 \times \frac{1}{16} + 4 \times \frac{1}{16^2} + 7 \times \frac{1}{16^3} + A \times \frac{1}{16^4} + E \times \frac{1}{16^5} + 1 \times \frac{1}{16^6} = 0.32999996 \approx 0.33$$

## 3 bc 程序用法

linux 系统有一个命令 bc，是一个命令行的计算软件。下面演示用 bc 算 0.33 的二进制。

### 3.1

下面的演示中, " \$" 开头的表示在 shell 输入的命令, " //" 表示注释,

```
$ bc
```

进入 bc 后

```
输入 obase=2 // 以二进制显示计算结果
```

```
    0.333 // 输入十进制的 0.333
```

```
输出 .0101010100 // 0.333 的二进制
```