

补充文档0x1 进制于进制的转换

进制转换，是计算机科学的一个非常基础的知识，在搞清楚进制转换之前我们需要简单的理一理什么叫进制

tips:在这篇文章中我们用 $(X)_n$ 的形式表示n进制的数字，比如 $(10)_{10}$ 表示十进制的数字10，当然，你在Play with Python 0x04_0x01【Numbers —— 数字】章节看到的【0b,0x,0o】的写法，在计算机科学中也是通用的，大部分的编程语言和操作系统都是支持这种写法的，这里纯粹用数学的表达法来写，更加的广义(比如7进制似乎在计算机科学中就没有被定义)

任意进制 转换 十进制

我们人有两只手，一只手上五个指头，所以数数的时候，我们习惯从1数到10，或者从0数到9，但是不论是从0到9还是从1到10，都一共是10个数字，但是10是一个两位数，所以我们论述的时候都说从0~9

ok，从0~9十个数字

从0~9，大于9就进一，十进制里，单个的位上最大的数字就是9

到这里为止没什么问题，的确这是小学的内容

然后我们看一点稍微进阶一些的内容

十进制数字9999，可以分解成： $9 + 90 + 900 + 9000$ 对吧？

- $(9)_{10}$ ，是 $10^0 * 9$ ，也就是所谓的个位
- $(90)_{10}$ 是 $10^1 * 9$ ，也就是所谓的十位
- $(900)_{10}$ 是 $10^2 * 9$ ，也就是所谓的百位
- $(9000)_{10}$ 是 $10^3 * 9$ ，也就是所谓的千位

所以

$$\begin{aligned} 9999 &= 10^0 * 9 + 10^1 * 9 + 10^2 * 9 + 10^3 * 9 \\ &= 9 + 90 + 900 + 9000 \end{aligned}$$

到此为止我们总结出了一个比较简单的“规律”

即：

1. n进制的最大数字是n-1
2. n进制的最小数字我们一般约定为0
3. n进制的数字可以转换成 $\sum_n^1 n^n * k$ 的形式

所以以此类推，二进制的最大数字就是 $2 - 1$ ，就是1，最小数字就是0

并且呢，根据上述的【3】，我们可以把一个二进制的数字进行转换：

$$(1011)_2 = 2^0 * 1 + 2^1 * 1 + 2^2 * 0 + 2^3 * 1 = 1 + 2 + 0 + 8 = 11$$

这就是从2进制转换成10进制的方法

把数字 $(X)_2$ 的每一位按照

- 位对应的 $2^n * \text{位上的数字}$ (1或者0)

的方式，就可以把一个二进制的数字转化为10进制的数字

那么8进制呢？

$$\begin{aligned}
 (165)_8 &= 8^0 * 5 + 8^1 * 6 + 8^2 * 1 \\
 &= (5 + 48 + 64)_{10} \\
 &= (117)_{10}
 \end{aligned}$$

以此类推

但是到了16进制就不太一样了

根据刚刚说的，16进制的最大数字应该是15，但是我们的数字系统已经没有对应的数字了

怎么办呢？

使用字母就可以

16进制的数字，使用0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *A, B, C, D, E, F*

来表示，*A - F*表示10 - 15

那么问题来了，我们十个手指，最大能表示多少呢？

按照二进制的思想走就是 $(1111111111)_2$ ，也就是 $(1023)_{10}$

其实据说中国古人也有用12进制的

伸出你的手，那只手都行，除去大拇指的四个指头，每个指头3个指节，一共12个指节，用大拇指点着指节转着圈数，一样可以计数，但是不论如何，读到这里的你来说，任何进制转换成10进制都不在话下了吧？

那么如何从10进制往外转换呢？

$(1234)_{10} \rightarrow (X)_2$

这个算法也很简单