计算机算数

9.1十进制系统

9.2位数字系统

9.3二进制系统

9.4二进制与十进制之间的转换

整数

分数

9.5十六进制表示法

9.6 关键术语和习题

**学习目标**

学习本章后，你应该能够：

* 理解**位数系统**的基本概念和术语。
* 描述对整数和小数进行**二进制**和**十进制**转换的方法。
* 描述**十六进制表示法**的基本原理。

## 9.1十进制系统

在日常生活中，我们使用基于十进制数字（0，1，2，3，4，5，6，7，8，9）的系统来表示数字，我们将该系统称为十进制系统。想一想数字83意味着什么，它意味着八个十加上三：

83 =（8×10）+ 3

数字4728表示四个一千、七个一百、二个十，加上八：

4728 =（4×1000）+（7×100）+（2×10）+8

十进制以10的**基（base）**或**基数（radix）**。十进制意味着数字的大小等于每一位的数字乘上10对应权值的幂次，如：

83 =（8×101）+（3×100）

4728 =（4×103）+（7×102）+（2×101）+（8×100）

同样的原理也适用于小数部分，但是使用的是10的负幂次。例如，小数0.256代表2/10加上5/100，再加上6/1000：

0.256 =（2×10-1）+（­5×10-2）+（6×10-3）

同时具有整数部分和小数部分的数字则同时用的10的正次幂和负次幂，例如:

442.256 =（4×102）+（4×101）+（2×100）+（2×10-1）+（5×10-2）+（6×10-3）

任何数字中，最左边的数字被称为**最高有效数字**，因为它有最高权值。最右边的数字称为**最低有效数字**。在刚刚提到的十进制数中，左边的4是最高有效数字，右边的6是最最低有效数字。

表9.1显示了数字的每一位与分配给该位的值之间的关系。每一位的权值是该位相邻右边位的10倍，是相邻左边位的十分之一。因此，数位的权值为10的连续幂次。如果我们按照表9.1对位置进行编号，那么位置i被的权值为10i。



一般来说，由X = {…d2d1d0.d-1d-2d-3…}使用十进制表示法所表示的数字为：

X = Σ(di ×10i)

还有一个值得思考的点：不妨考虑，在数字509中有多少个10。由于在十位上的数字是0，你可能会因此认为没有一个10。但实际上该数字有50个10。十位上的0意味着该数字十位进位到高位后没有数字剩下。因此，由于每一位的数字是该位进位到高位后剩下的数字，所以每一位的值都不能大于9。9是每一位进位到相邻高位前的最大数字。

## 9.2位数字系统

在位数字系统中，每个数字由一串数字表示，其中每个数字位置i具有相应的权重ri，其中r是数字系统的基数或基。在具有基r的系统中，数字的一般形式是

(…a3a2a1a0.a-1a-2a-3…)r

其中每一位ai的值是介于0到r的整数。a0和a-1之间的点称为**基点**。数值被定义为:

… + a3r3 + a2r2+ a1r1 + a0r0+ a-1r-1 + a-2r-2 + a-3r-3 + …

= Σ(ai × bi) (9.2)

那么，十进制是基数为10、数字范围为0到9的位数系统的特殊情况。

另外考虑一个以7为基的另一个位数字系统。表9.2显示了从位置-1至位置4的加权值。每个位置上的数字范围为从0到6。



## 9.3二进制系统

在小数系统中，使用10种不同的数字来表示以10为基的数字。在二进制系统中，我们只使用两个数字，1和0。因此，二进制系统中的数字的基为2。

为了避免混淆，我们有时会在数字上加一个下标来表示它的基数。例如，8310和472810是用十进制符号表示的数字，或者更简单地说，是十进制数字。二进制符号中的数字1和0具有与十进制符号相同的含义：

02 = 010

12 = 110

为了表示更大的数字，如同十进制符号一样，二进制数字中的每个数字都有一个与其位置相关的权值，如下：

102 =（1\* 21）+（0\* 20) = 210

112 =（1\* 21）+（1\* 20）= 310

1002 =（1\* 22)+（0\* 21）+（0\* 20）= 410

等等。同样，小数值用基数的负幂次来表示：

1001.101 ＝ 23＋20＋2-1 + 2-3= 9.62510

一般来说，由 Y = {…b2b1b0.b-1b-2b-3…} 使用二进制表示法所表示的数字为：

Y = Σ(bi ×2i) (9.3)

## 9.4二进制与十进制之间的转换

将数字从二进制转换为十进制很简单。事实上，我们在前一小节中已经举了几个例子。只需要将每个二进制数字乘以2的适当幂次并将乘积相加。

将十进制小数转换为二进制时，需要将整数部分和小数部分分别进行处理。

### 整数

对于整数部分，回想一下，在二进制符号中，表示为

Bm-1bm-2…b2b1b0 bi = 0或1

其十进制值为：

（bm-1×2m-1）+（bm-2× 2m-2）+ … +（b1× 21）+ b0

假设现在需要将十进制整数N转换为二进制形式。如果我们在十进制制中把N除以2，得到商N1和余数R0，我们可以写成:

N ＝ 2 × N1 + R0 R0 = 0 或1

然后，我们将商N1再除以2。假设新得到的商是N2，余数是R1。也就是：

N1 ＝ 2 × N2 + R1 R1 = 0或1

整理上面的式子：

N ＝ 2 × N1 + R0 = (N2 ×22)+ (R1 ×21) + R0

如果还有：

N2 = 2N3 + R2

则可得：

N = (N3 ×23)+ (R2 ×22) + (R1 ×21) + R0

因为N > N1 > N2 …，所以继续这个序列推到最终将产生一个商Nm-1 = 1（除了十进制整数0和1，它们的二进制表示分别是0和1）和一个余数Rm-2，即0或1。那么可得：

N ＝(1×2m-1) + (Rm-2×2m-2) + ... + (N2×22)+ (R1×21) + R0

上式是N的二进制表达式。从而，我们通过连续除以2的方法将以10为基转换为以2为基。余数和最终的商1，按照幂次递增的顺序写下来，就得到了N的二进制表示。图9.1给出了两个例子。

### 小数

对于小数部分，回想一下，在二进制符号中，值介于0和1之间的数字可以表示为

0.b-1b-2b-3… bi = 0 或 1

其大小为：

(b-1×2-1) + (­b-2×10-2) + (b-3×10-3)...

此式子可以改写为：

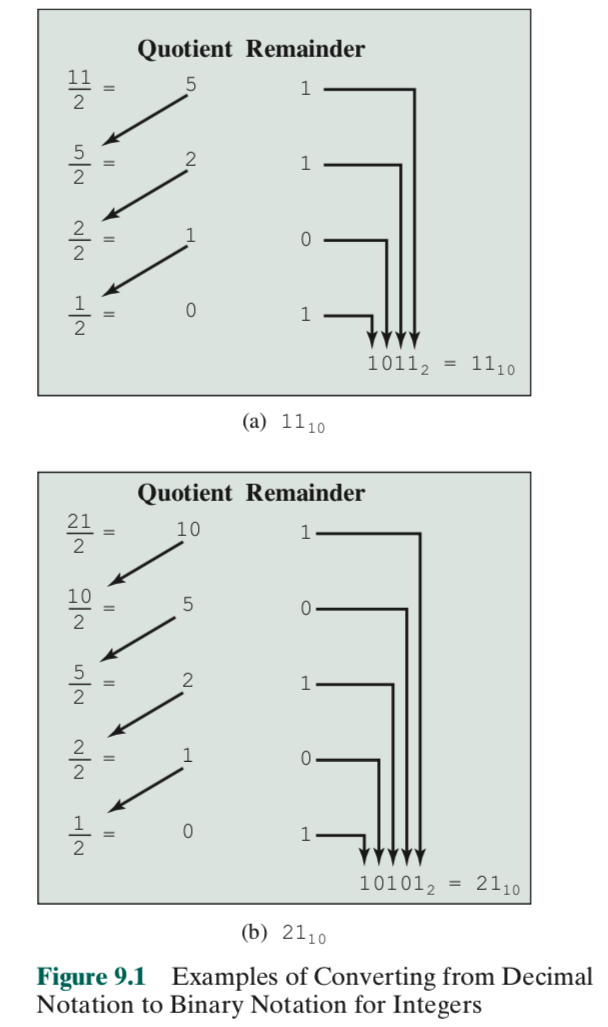
2-1×(b-1 + 2-1×(b-2 + 2-1×(b-3 + …)…))

这个表达使用了一种转化的技巧。假设我们想把数字F(0 < F < 1)从十进制转换为二进制，F可以用这种形式来表达：

F = 2-1×(b-1 + 2-1×(b-2 + 2-1×(b-3 + …)…))

如果我们在等式两边同时乘以2，可以得到：

2×F = b-1 + 2-1×(b-2 + 2-1×(b-3 + …)…)

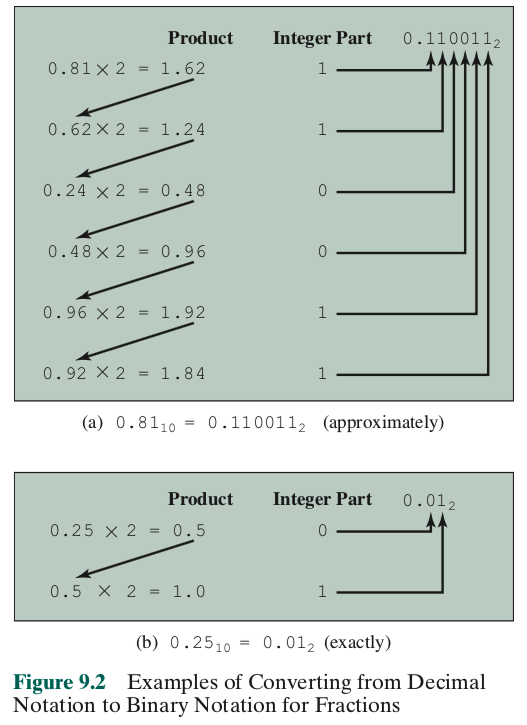


从这个等式中，我们看到（2×F）的整数部分，只能是0或1，因为0< F1 <1，所以就是b-1。因此我们可以说(2×F) = b-1 + F1，其中0 < F1 < 1且

F1 = 2-1×(b-2 + 2-1×(b-3 + 2-1×(b-4 + …)…))

为了找到b-2，我们重复以上过程。因此，转换算法需要连续乘以2，执行每一步时，来自前一步的数字的小数部分会乘以2。乘积中小数点左边的数字总是0或1，从最高有效数字开始可以使用二进制表示出此数字。在下一步中，乘积的小数部分会被用作乘数。图9.2显示了两个例子。

这个过程并不一定精确，也就是说，具有有限个数字的十进制小数可能需要具有无限个数字的二进制小数来表示。在这种情况下，转换算法将会根据期望的精度，在执行完预先设定数量的步骤之后停止。

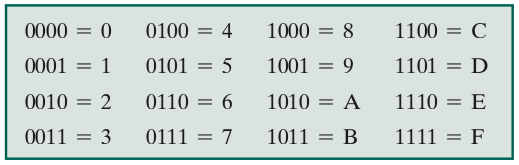


### 9.5十六进制表示法

由于数字计算机部件固有的二进制特性，计算机内的所有形式的数据都是由各种各样的二进制代码表示的。然而，无论二进制系统对于计算机多么方便，对于人类来说都非常繁琐。因此，对于必须花时间处理计算机中实际原始数据的大多数计算机专业人员来说，他们更喜欢更紧凑的数字表示法。

使用什么进制呢？一种可能的选择是十进制表示法。这当然比二进制符号更紧凑，但是尴尬的是在基数2和基数10之间进行转换十分繁琐。

取而代之的是一种称为十六进制的表示法。二进制数字被分成四位一组的小集合，被称为**半字节**。给四个二进制数字的每种可能的组合一个对应的标识符，如下所示：



因为使用了16个符号，所以这种表示方法被称为**十六进制**，这16个符号被称为**十六进制数字**。

十六进制数字的序列可以用来表示基数为16的整数（如表9.3所示）。因此，

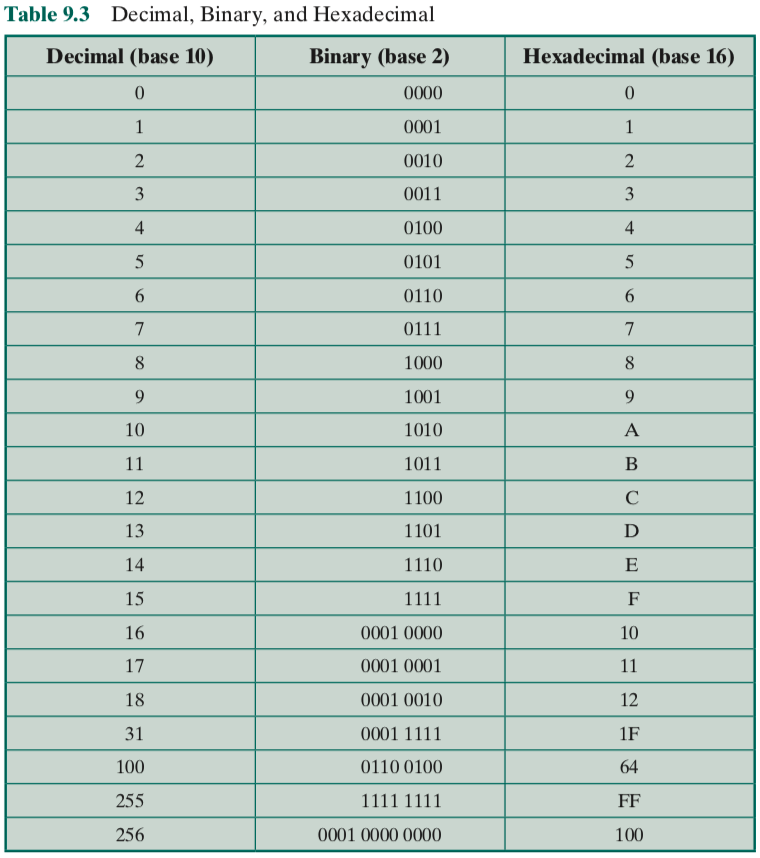
2C16 = (216×161) + (C16×160)

= (210×161) + (1210×160) = 44

因此，如果将十六进制数视为以16为基的位数字系统中的数字，我们有

Z = Σ(hi ×16i) (9.4)

其中16是基数，每个十六进制数字hi在十进制下范围为0≤hi＜15，相当于十六进制下范围0≤hi≤F。



十六进制符号不仅可以用来表示整数，而且可以作为简明符号来表示任何二进制数字序列，不管它们是表示文本、数字还是表示其他类型的数据。使用十六进制记数法的原因如下：

1. 它比二进制表示更紧凑。
2. 在大多数计算机中，二进制数据常常是4位的倍数，因而可以用多个十六进制数来表示。
3. 在二进制和十六进制符号之间进行转换是非常容易的。

作为最后一点的示例，考虑二进制字符串110111100001。它其实相当于

1101 1110 0001 = DE116

D E 1

这个过程执行得如此自然，以至于有经验的程序员可以在头脑中直接将写好的二进制数据转换成十六进制等值数据，而无需动笔。

### 9.6关键术语和习题

**关键术语**

|  |  |  |
| --- | --- | --- |
| base 基 | hexadecimal 十六进制 | nibble 半字节 |
| binary 二进制 | integer 整数 | positional number system 位数字系统 |
| decimal 十进制 | least significant digit 最低有效数字 | radix 基数 |
| fraction 分数 | most significant digit 最高有效数字 | radix point 小数点 |

**习题**

9.1 在以下面数字为基数的情况下，从1写到2010:

a.8 b.6 c.5 d.3

9.2 将数字(1.1)2、(1.4)10和(1.5)16从小到大排序。

9.3 按照指定的方向进制转换：

a.548转换为5进制 b.3124转换为7进制 c.5206转换为7进制d转换为5进制.122123转换为9进制

9.4 将数字从一个基数转换为该基数的幂次数，有什么共同特点？例如，从以3为基数转化为以9（32）为基数，或从以2为基数转化为以4（22）为基数或以8（23）为基数。

9.5 将下列二进制数转换为十进制数：

a. 010101 b. 110011 c.100011 d.101000 e.101100

9.6 将下列二进制数转换为十进制数：

a. 001100.101 b.010010.111 c.0010111111.11

9.7 将下列十进制数转换为二进制数：

a. 27 b. 81 c. 126 d.207 e. 333

9.8 将下列十进制数转换为二进制数：

a. 36.25 b.72.75 c. 45.3750

9.9 证明具有终止二进制（二进制小数点右边有有限个数字）表示的每个实数，也有终止十进制表示（小数点右边有有限个数字）。

9.10 用十六进制表示法表示下列八进制数（以8为基数的数）：

a.12 b.5655 c.2550276 d.76545336 e.3726755

9.11 将下列十六进制数转换为十进制数：

a.AA b.18D c.B9 d.45C e.BADE

9.12 将下列十六进制数转换为它们的十进制等效数：

a. C.8 b. A9.A c. 939.3 d. 147.1 e. BAD.E

9.13 将下列十进制数转换为十六进制数：

a. 45 b. 72 c. 3204 d. 6666 e. 14,050

9.14 将下列十进制数转换为十六进制数：

a. 204.125 b. 255.875 c.631.25 d.10000.00390625

9.15 将下列十六进制数转换为二进制数：

a. E b. 1C c. A64 d.1F.C e. 239.4

9.16 将下列二进制数转换为十六进制数：

a. 1001.1111 b.110101.011001 c.10100111.111011