第 2 章　计算机中信息的表示

教学目标

1. 了解计算机的信息表示形式
2. 掌握并熟悉计算机中各种数制间的相互转换
3. 熟悉计算机中编码的表示形式和编码类型

2.1　数制的概念

2.1.1　数制

1. 数制的概念

数制是用一组固定的数字和一套统一的规则来表示数目的方法。

按照进位方式计数的数制叫做进位计数制。在生活中有许多各种各样的进制计数法，例如：逢十进一即十进制，人类屈指计数沿袭至今且最为习惯；十二进制作为商业包装计量单位“一打”的计数方法；十六进制为中药或金器等采用的计量单位；60 分钟为 1 小时，用的是 60 进制计数法；一星期有 7 天，是七进制计数法；一年有 12 个月，是十二进制计数法等。

进位计数制的两个要素：

（1）基数（用 r 表示）：它是指各种进位计数制中允许选用基本数码的个数。例如，十进制的数码有：0、1、2、3、4、5、6、7、8 和 9，因此，十进制的基数为 10。

（2）位权：每个数码所表示的数值等于该数码乘以一个与数码所在位置相关的常数，这个常数叫做位权。其大小是以基数为底、数码所在位置的序号为指数的整数次幂。例如：十进制的位权为 10^i。

位权表示法的原则是数码的总个数等于基数；每个数码都要乘以基数的幂次，而该幂次是由每个数所在的位置所决定的。任意一个的 n 位整数部分和 m 位小数部分，排列方式是以小数点为界，整数自右向左数，依次取 i=1、2、3、…、n-1。小数点部分，依次取 i=-1、-2、-3、…、-m。如十进制数 634.28 可以标为：$6^2 3^1 4^0.0^{-1}8^{-2}$。

基数与位权的关系是：位权的值是基数的若干次幂。因此，用任何一种数制表示的数都可以写成按位权展开的多项式之和。

如：$(634.08)_{10}=6\times10^2+3\times10^1+4\times10^0+0\times10^{-1}+8\times(10)^{-2}=634.08$

2. 几种常用进制及其特点（如表 2-1 所示）

（1）十进制（Decimal notation）。

1）十进制基本特点。

①十个数码：0，1，2，3，4，5，6，7，8，9。

②逢十进一，借一当十。

2）十进制数按权展开式。

任意一个 n 位整数和 m 位小数的十进制数 D 可表示为：

$D＝D_{n-1}\times10^{n-1}+D_{n-2}\times10^{n-2}+\cdots+D_0\times10^0+D_{-1}\times10^{-1}+\cdots+D_{-m}\times10^{-m}$

（2）二进制（Binary notation）。

1）二进制基本特点。

①两个数码：0，1。

②逢二进一，借一当二。

2）二进制数按权展开式。

任意一个 n 位整数和 m 位小数的二进制数 B 可表示为：

$B＝B_{n-1}\times2^{n-1}+B_{n-2}\times2^{n-2}+\cdots+B_0\times2^0+B_{-1}\times2^{-1}+\cdots+B_{-m}\times2^{-m}$

（3）八进制（Octal notation）。

1）八进制基本特点。

①八个数码：0，1，2，3，4，5，6，7。

②逢八进一，借一当八。

2）八进制数按权展开式。

任意一个 n 位整数和 m 位小数的八进制数 Q 可表示为：

$Q＝Q_{n-1}\times8^{n-1}+Q_{n-2}\times8^{n-2}+\cdots+Q_0\times8^0+Q_{-1}\times8^{-1}+\cdots+Q_{-m}\times8^{-m}$

（4）十六进制（Hexdecimal notation）。

1）十六进制基本特点。

①十六个数码：0，1，2，3，4，5，6，7，8，9，A，B，C，D，E，F。

在十六个数码中的 A、B、C、D、E 和 F 六个数码，分别代表十进制数中的 10、11、12、13、14 和 15，这是国际上通用的表示法。

②逢十六进一，借一当十六。

2）十六进制数按权展开式。

任意一个 n 位整数和 m 位小数的十六进制数 Q 可表示为：

$H＝H_{n-1}\times16^{n-1}+H_{n-2}\times16^{n-2}+\cdots+H_0\times16^0+H_{-1}\times16^{-1}+\cdots+H_{-m}\times16^{-m}$

表 2-1　计算机中常用的几种进制数的表示

进位制	二进制	八进制	十进制	十六进制
规则	逢二进一，借一当二	逢八进一，借一当八	逢十进一，借一当十	逢十六进一，借一当十六
基数	r=2	r=8	r=10	r=16
数码	由数字 0、1 组成	由数字 0～7 组成	由数字 0～9 组成	由数字 0～9、A～F 组成
位权	2^i	8^i	10^i	16^i
形式表示	B（Binary）	O（Octal）	D（Decimal）或不加字母	H（Hexadecimal）

（5）常用数制的对应关系表，如表 2-2 所示。

表 2-2 常用数制的对应关系表

二进制数	十进制数	八进制数	十六进制数
0	0	0	0
1	1	1	1
10	2	2	2
11	3	3	3
100	4	4	4
101	5	5	5
110	6	6	6
111	7	7	7
1000	8	10	8
1001	9	11	9
1010	10	12	A
1011	11	13	B
1100	12	14	C
1101	13	15	D
1110	14	16	E
1111	15	17	F
10000	16	20	10

3. 计数制的书写规则

（1）在数字后面加写相应的英文字母作为标识。

如：二进制数的 100 可写成 100B，十六进制数 100 可写成 100H。

（2）在括号外面加数字下标。

如：$(1011)_2$ 表示二进制数 1011，$(2DF2)_{16}$ 表示十六进制数 2DF2。

2.1.2 二进制数据表示

计算机中电子器件来存储信息等都是用二进制进行编码的，即信息表示。二进制并不符合人们的习惯，但是计算机内部仍采用二进制表示信息，其主要原因有以下几点：

（1）电路简单。计算机是由逻辑电路组成的，逻辑电路通常只有两个状态，例如晶体管的饱和与截止、开关的接通与断开、电压电平的高与低等。这两种状态正好用来表示二进制数的两个数码 0 和 1。

（2）可靠性高。二进制数的每一位只有 0 和 1 两状态，只需要两种设备就能表示，所以二进制数节省设备。由于状态简单，所以抗干扰力强，可靠性高。

（3）运算简单。二进制运算法则简单。

（4）逻辑性强。计算机工作原理是建立在逻辑运算基础上的，逻辑代数是逻辑运算的理论依据。二进制只有两个数码，正好代表逻辑代数中的“真”和“假”。

二进制的主要缺点是数位太长，不便阅读和书写，人们也不习惯。为此常用八进制和十

六进制作为二进制的缩写方式。为了适应人们的习惯，通常在计算机内都采用二进制数，输入和输出采用十进制数，由计算机自己完成二进制与十进制之间的相互转换。在输入过程中，系统自动将用户输入的各种数据按编码的类型转换成相应的二进制形式存入计算机存储单元中。在输出过程中，再由系统自动将二进制编码数据转换成用户可以识别的数据格式输出给用户。

二进制数的算术四则运算规则，除进、借位外与十进制数相同。

（1）二进制加法规则。

0+0=0　 1+0=1

0+1=1　 1+1=10（逢二进一，向高位进一位）

（2）二进制减法规则。

0-0=0　 0-1=-1（向高位借一位，借一当二）

1-0=1　 1-1=0

（3）二进制乘法规则。

0×0=0　 1×0=0

0×1=0　 1×1=1

（4）二进制除法规则。

0÷0=0　 1÷0（无意义）

0÷1=0　 1÷1=1

2.2　数制转换

将数由一种数制转换成另一种数制称为数制间的转换。由于计算机采用二进制，但用计算机解决实际问题时对数值的输入输出通常使用十进制，这就有一个十进制向二进制转换或由二进制向十进制转换的过程。也就是说，在使用计算机进行数据处理时首先必须把输入的十进制数转换成计算机所能接受的二进制数；计算机在运行结束后，再把二进制数转换为人们所习惯的十进制数输出。这两个转换过程完全由计算机系统自动完成，不需人们参与。

虽然在计算机内部使用二进制数进行工作，但对于广大用户来说，使用二进制数是很不方便的。因为二进制数的数位比起等值的十进制数要长得多，而且读写也比较困难，所以人们通常使用八进制和十六进制作为二进制的缩写方式。这样就存在了一个不同进制之间的转换问题。

不同进制之间的转换都要遵守的规则是：将整数部分和小数部分分别进行转换，然后用小数点连接。

2.2.1　非十进制数转换为十进制数

由于任一数都可以按权展开，于是很容易将一个非十进制数转换为相应的十进制数。具体的步骤是：将一个非十进制按权展开成一个多项式，每项是该位的数码与相应的权之积，把多项式按十进制数的规则进行计算机求和，所得结果即是该数的十进制。

（1）二进制数转换成十进制数。二进制数转换成十进制数只需按权展开然后相加即可，即按权相加法。把第一位的权（2 的某次幂）与数位值（0 或 1）的乘积相加，其和就是相应的十进制数。

例如：$(101.1)_2=1\times2^2+0\times2^1+1\times2^0+1\times2^{-1}=(5.5)_{10}$

（2）八进制数转换成十进制数。八进制数转换成十进制数只需按权展开然后相加即可，即按权相加法。把第一位的权（8 的某次幂）与数位值的乘积相加，其和就是相应的十进制数。

例如：$(265)_8=2\times8^2+6\times8^1+5\times8^0=(181)_{10}$

（3）十六进制数转换成十进制数。十六进制数转换成十进制数只需按权展开然后相加即可，即按权相加法。把第一位的权（16 的某次幂）与数位值的乘积相加，其和就是相应的十进制数。

例如：$(B5)_{16}=11\times16^1+5\times16^0=(181)_{10}$

2.2.2 十进制数转换为非十进制数

在十进制数转换为非十进制数的过程中，要按照十进制数的整数部分和小数部分分两种不同的情况采用不同的方法来处理。

- 十进制整数转换成非十进制整数时，采用余数法：用十进制整数除基数，当商是 0 时，将余数由下而上排列。
- 十进制小数转换成非十进制小数时，采用进位法：用十进制小数乘基数，当积值为 0 或达到所要求的精度时，将整数部分由上而下排列。

（1）十进制数转换成二进制数。十进制数有整数和小数两部分，转换时整数部分采用“除二取余”法，小数部分采用“乘二取整”法，然后通过小数点将转换后的二进制数连接起来即可。

十进制整数转换成二进制采用“除二取余”的方法，即将十进制整数除以 2，得到一个商和余数，再将商除以 2，得到另一个商和余数，如此继续下去，直到商为 0 为止。最后将余数（均为 0 或 1）按逆向的方式（即最后一个余数为二进制数的最高位，第一个余数为二进制数的最低位）依次排列，即得到所求二进制数的各位数字。

十进制纯小数转换为二进制数时采用“乘二取整”的方法，即将十进制纯小数乘以 2，得到一个积，然后去掉积的整数部分，将剩下的纯小数再乘以 2，如此继续下去，直到纯小数部分为零或满足所要求的精度为止。最后将去掉的整数部分（0 或 1）按乘得的先后依次排列下去，即得所求二进制纯小数的小数点后各位数字。

混合小数由整数和纯小数部分组成，在进行十进制混合小数转换为二进制数时，将这两部分分别按前面所介绍的方法分别转换为对应的二进制整数与二进制小数，然后再用小数点组合起来即可。

（2）弄清二进制数与十进制数的互换方法，可将其推广到其他进制与十进制数的互换，不同之处是应该考虑具体进制的基数，而转换算法完全是一样的。十进制数转换成八进制数采用“除八取余”法（整数部分）与“乘八取整”法（小数部分），十进制数转换成十六进制数采用“除十六取余”法（整数部分）与“乘十六取整”法（小数部分）。

2.2.3 十进制数与任意进制数之间转换举例

例如：将$(35.6875)_{10}$转换为二进制数。

①用“除二取余”法将整数部分$(35)_{10}$转换为二进制整数：

2	35		
2	17	…… 余数为 1	↑（低位）
2	8	…… 余数为 1	
2	4	…… 余数为 0	
2	2	…… 余数为 0	
2	1	…… 余数为 0	
	0	…… 余数为 1	（高位）

故：$(35)_{10} = (100011)_2$

验证：$1\times2^5+0\times2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0=32+2+1=35$

②用“乘二取整”法将小数部分$(0.6875)_{10}$转换为二进制形式：

0.6875 × 2 = 1.375	…… 整数部分为 1	高位 ↓
0.3750 × 2 = 0.75	…… 整数部分为 0	
0.75 × 2 = 1.5	…… 整数部分为 1	
0.5000 × 2 = 1.0	…… 整数部分为 1	低位

即：$(0.6875)_{10} = (0.1011)_2$

③ 整数部分与小数部分合并，可得：$(35.6875)_{10} = (100011.1011)_2$

注意：在上例中，将十进制小数转换成为二进制小数的过程中，乘积小数部分变成“0”，表明转换结束。实际上将十进制小数转换成二进制、八进制、十六进制小数过程中小数部分可能始终不为零，因此只能限定保留若干位为止。

将十进制数转换为八进制、十六进制数的规则和方法与之相同，只是 r（基数）的取值不同。

2.2.4　二进制数与八进制数及十六进制数之间的转换

十进制、八进制、十六进制之间的转换，比较难计算，而且一般用得很少，最常用的方法是将其转换成二进制，然后再转换一次。

（1）二进制数与八进制数的互换。因二进制数基数是 2，八进制数基数是 8。又由于 2^3 ＝8，8^1＝8，从常用数制的对应关系表中可以看到 3 位二进制数恰好是一位八进制数，所以二进制与八进制互换是十分简便的。

1）二进制数转换成八进制数。二进制数转换为八进制数可概括为“三位一组”，即以小数点为基准，整数部分从右至左，每三位一组，最高位不足三位时，添 0 补足三位；小数部分从左至右，每三位一组，最低有效位不足三位时，添 0 补足三位。然后将各组的三位二进制数

按权展开后相加，得到一位八进制数码。再按权的顺序连接起来即得到相应的八进制数。

例如，将$(1011100.00111)_2$转换为八进制数：

$(001,011,100.001,110)_2=(134.16)_8$

1　3　4 . 1　6

2）八进制数转换成二进制数。八进制数转换成二进制数可概括为“一位拆三位”，即把一位八进制数写成对应的三位二进制数，然后按权连接即可。

例如，将$(163.54)_8$转换成二进制数：

$(1\quad 6\quad 3\,.\,5\quad 4\,)_8=(1110011.1011)_2$

001,110,011.101,100

（2）二进制数与十六进制数的互换。二进制数与十六进制数之间也存在二进制数与八进制数之间相似的关系。由于 $2^4=16$，$16^1=16$，即二进制四位数对应于十六进制一位数。从常用数制的对应关系表中也能看到一位十六进制数恰好对应四位二进制数，所以二进制数与十六进制数之间的转换同二进制数与八进制数之间的转换相仿，只是按四位二进制数对应一位十六进制数进行分组的。

1）二进制数转换成十六进制数。二进制数转换为十六进制数可概括为“四位一组”。即以小数点为基准，整数部分从右至左，小数部分从左至右，每四位一组，不足四位添 0 补足。然后将每组的四位二进制数按权展开后相加，得到一位十六进制数码，再按权的顺序连接起来即得到相应的十六进制数。

例如，将$(1011100.00111)_2$转换为十六进制数：

$(0101,1100.0011,1000)_2=(5C.38)_{16}$

5　C .　3　8

2）十六进制数转换成二进制数。十六进制数转换成二进制数可概括为“一位拆四位”，即把一位十六进制数写成对应的四位二进制数，然后按权连接即可。

例如，将$(16E.5F)_{16}$转换成二进制数：

$(1\quad 6\quad E\,.\quad 5\quad F\,)_{16}=(101101110.01011111)_2$

0001,0110,1110.0101,1111

2.3　信息编码

计算机是以二进制方式组织、存放信息的，信息编码就是指对输入到计算机中的各种数据用二进制数进行编码的方式。数据是指能够输入计算机并被计算机处理的数字、字母和符号的集合。平常所看到的景象和听到的事实，都可以用数据来描述。可以说，只要计算机能够接受的信息都可叫数据。计算机中的数据分为两大类：数值型数据和非数值型数据。字符型数据就属于典型的非数值型数据。对于不同机器、不同类型的数据，其编码方式是不同的，编码的方法也很多。为了使信息的表示、交换、存储或加工处理方便，在计算机系统中通常采用统一的编码方式，因此制定了编码的国家标准或国际标准，如位数不等的二进制码、十进制码、BCD 码、ASCII 码、汉字编码等。计算机使用这些编码在计算机内部和键盘等终端之间以及计算机之间进行信息交换。

2.3.1　信息存储的单位

数据在计算机中都必须以二进制形式表示。一串二进制数既可表示数量值，也可表示一个字符、汉字或其他。一串二进制数代表的数据不同，含义也不同。这些数据在计算机的存储设备中是如何进行组织存储的？

1. 位（bit）

位是计算机存储数据的最小单位。一个二进制位只能表示 0 或 1 两种状态，要想表示更多的信息，就得把多个位组合起来作为一个整体，一般以 8 位二进制组成一个基本单位。

2. 字节（Byte）

字节是计算机数据处理的基本单位，即以字节为单位存储和解释信息。规定一个字节等于 8 位二进制，即 1B＝8bit。通常，1 个字节可存放一个 ASCII 码，2 个字节存放一个汉字国标码，整数用 2 个字节组织存储，单精度实数用 4 个字节组织成浮点形式，而双精度实数利用 8 个字节组织成浮点形式，等等。

存储器容量大小是以字节数来度量，经常使用三种度量单位，即 KB、MB 和 GB，其大小分别为：

$1KB = 2^{10} = 1024B$

$1MB = 2^{10} \times 2^{10} = 1024 \times 1024 = 1048576B$

$1GB = 2^{10} \times 2^{10} \times 2^{10} = 1024 \times 1024 \times 1024 = 1073741824B$

3. 字（Word）

计算机处理数据时，CPU 通过数据总线一次存取、加工和传送的数据长度称为字。一个字通常由一个字节和若干字节组成。由于字长是计算机一次所能处理的实际位数长度，所以字长是衡量计算机性能的一个重要标志，决定了计算机数据处理的速度，字长越长，性能越强。

不同的计算机字长是不相同的，常用的字长有 8 位、16 位、32 位、64 位不等。

2.3.2　数值型数据的编码

数值型数据是指可进行算术运算的数据，具有量的含义，且有正负之分、整数和小数之分，如(258)10，(0101.0110)10，3AH 等都是数值型数据。

在计算机中表示一个数值型数据，首先需要确定数的长度（字长）、符号（正、负数）和小数点的表示。

（1）字长：在我们熟悉的十进制中，数的长度表示为位数，例如 118 为三位数，12321 为五位数。在计算机中数据长度也称为字长，按字节来计算，并与存储容量的计量单位“字节”一致。应该指出的是，数学中数的长度长短可以不一致，该是多少位就是多少位。但在计算机中，如果数据的长度也随数而异，长短不一，无论存储或处理都会十分不便，所以在同一计算机中，数据的长度常常是统一的。当长度不一致时，用“0”填之，即同类数据使用相同的数据长度，且与表示数值大小的实际长度（二进制位数）无关。

（2）正、负数的表示：数有正、负之分，在计算机中总是用数的最高位（左边第一位）来表示数的符号，并约定以“0”代表正数，以“1”代表负数。

（3）小数点位置确定。在计算机中表示数值型数据，小数点的位置总是隐含的，以便节省存储空间。隐含的小数点位置可以是固定的，也可以是可变的。前者称为定点数（Fixed-point

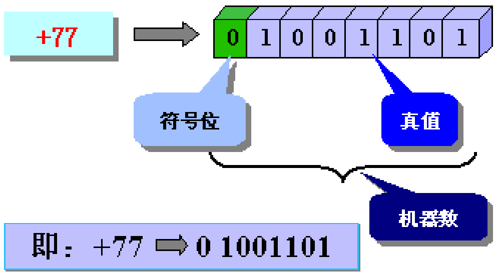
Number），后者称为浮点数（Floating-point Number）。在计算机中有两种数的表示方法，即定点表示法和浮点表示法。所谓定点表示法，就是小数点在数中的位置是固定不变的；所谓浮点表示法就是小数点在数中的位置是不不固定的，或者说是浮动的。

任何一个非二进制整数输入到计算机中都必须以二进制格式存放在计算机的存储器中，且用最高位作为数值的符号位，并规定二进制数“0”表示正数，二进制数“1”表示负数，每个数据占用一个或多个字节。这种连同数字与符号组合在一起的二进制数称为机器数，由机器数所表示的实际值称为真值。

1. *真值与机器数*

计算机机读数据采用二进制数的形式，对于带正、负符号的数值型数据，其符号也必须用二进制表示。对于带符号的数据，规定其最高位为符号位，且用“0”表示正，用“1”表示负，其他各位仍表示其数值。这种将符号数码化，并与数字合为一体的机内数表示形式，称为机器数。而它真正表示的数的数值称为这个机器数的真值。

例如：计算机中用 8 位二进制分别表示数+77。



2. *机器数的表示方法*

在计算机中，机器数也有不同的表示方法，通常用原码、反码和补码三种方式表示，其主要目的是解决减法运算。任何正数的原码、反码和补码的形式完全相同，负数则各自有不同的表示形式。

（1）原码。

规则：符号位“0”表示正数，“1”表示负数。其数值部分就是整数 X 的绝对值。

例如：[-5]原=10000101

在原码中，0 有两种表示：

[+0]=00000000

[-0]=10000000

特点：

- 直观，与真值转换很方便。
- 进行乘、除运算方便。
- 加、减运算比较麻烦，比如：一个正数和一个负数相加必须要考虑符号问题。

（2）反码。

规则：对于正数，其反码与原码相同，对于负数，符号位为 1，其数值位 X 的绝对值取反

例如：[+2] 反=00000010

[-2] 反=11111101

[+0] 反=00000000

[-0] 反=11111111

（3）补码。

规则：正数，其反码与原码相同，负数，符号位为 1，其数值位按位“求反”再加 1。

例如：[-5]原=10000101

[-5]补=11111010+1=11111011

[+0]补=00000000

[-0]补=11111111+1=00000000

补码的性质：

①[X+Y]补＝[X]补+[Y]补，即两数之和的补码等于各自补码的和。

②[X-Y]补＝[X]补+[-Y]补，即两数之差的补码等于被减数的补码与减数相反数的补码之和。

③[[X]补]补＝[X]原，即按求补的方法，对[X]补再求一次，结果等于[X]原。

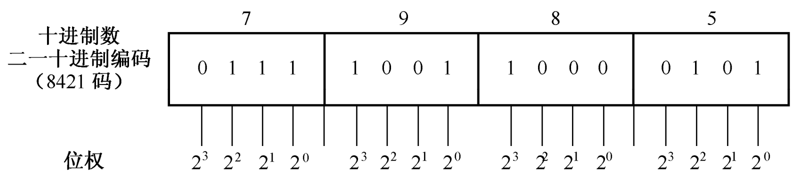
利用补码进行加、减运算：引进补码的目的是方便带符号数的加、减运算。

3. 数值数据的编码

在计算机中，对数字的输入和输出是用十进制数进行，而在计算机内部十进制数要用二进制编码来表示。凡采用若干位二进制数码表示一位十进制数的编码，统称为二进制编码的十进制数，也就是 BCD 码（Binary Coded Decimal），简称二－十进制编码。

二－十进制编码的方法很多，8421 码是最常用的一种，它采用 4 位二进制数表示 1 位十进制数，即每一位十进制数用四位二进制编码来表示。这 4 位二进制数各位权由高到低分别是 23、22、21、20，即 8、4、2、1。

例如：十进制数 7985 的 8421 码为 0111100110000101，如下所示。



2.3.3　非数值型数据的编码

非数值型数据是指输入到计算机中的所有信息，没有量的含义，不参与算术运算，如字符串"湖北武汉"、"2017 年 4 月"、"Windows 98"等，都是非数值型数据。上述字符串中虽然含有数字 2001，9，98 等，但它们不能、也不需要进行算术运算。

1. 字符编码

字符是计算机中使用最多的非数值型数据，是人与计算机进行通信、交互的重要媒介。目前，国际上通用的且使用最广泛的字符有：十进制数字符号 0～9，大小写的英文字母，各种运算符、标点符号等，这些字符的个数不超过 128 个。为了便于计算机识别与处理，这些字符在计算机中是用二进制形式来表示的，通常称之为字符的二进制编码。

由于需要编码的字符不超过 128 个，因此，用七位二进制数就可以对这些字符进行编码。但为了方便，字符的二进制编码一般占八个二进制位，它正好占计算机存储器的一个字节。具

体的编码方法，即确定每一个字符的七位二进制代码。但目前国际上通用的是美国标准信息交换码（American Standanl Code for Information Interchange ），简称为 ASCII 码（取英文单词的第一个字母的组合）。用 ASCII 表示的字符称为 ASCII 码字符。表 2-3 是 ASCII 码编码表。

表 2-3　ASCII 码编码表

$b_7\ b_6\ b_5$ / $b_4\ b_3\ b_2\ b_1$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	、	p
0001	SOH	DC1	!	1	A	Q	a	Q
0010	STX	DC2	“	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	S
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	^	7	G	W	g	w
1000	BS	CAN	（	8	H	X	h	x
1001	HT	EM	）	9	I	Y	i	y
1010	LF	SUB	*	：	J	Z	j	z
1011	VT	ESC	+	；	K	[	k	{
1100	FF	FS	,	<	L	\	l	;
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SL	US	/	?	O	_	o	DEL

表中前 32 个与最后一个是不可打印的控制符号。

特别需要指出的是，十进制数字字符的 ASCII 码与它们的二进制值是有区别的。

例如，十进制数 3 的七位二进制数为 0000011，而十进制数字字符“3”的 ASCII 码为 $(0110011)_2=(33)_{16}=(51)_{10}$，由此可以看出，数值 3 与数字字符“3”在计算机中的表示是不一样的。数值 3 能表示数的大小，并可以参与数值运算；而数字字符“3”只是一个符号，它不能参与数值运算。

2. 汉字编码

具有悠久历史的汉字是中华民族文化的象征。世界上四分之一以上的人口使用汉字，因此，在计算机中汉字的应用占有十分重要的地位。计算机在处理汉字信息时也要将其转化为二进制代码，这就需要对汉字进行编码。例如，当你用计算机编辑一篇文章时，就需要将文章中汉字及各种符号输入计算机，并进行排版、显示或打印输出。因此，必须解决汉字的输入、存储、处理和输出等一系列技术问题。由于汉字比西文字符不仅数量多，而且字形复杂，所以用计算机处理汉字要比处理西文字符困难得多。汉字处理技术的关键是汉字编码问题。根据汉字处理过程中不同的要求，汉字编码可分为国际码、输入码、机内码和字形码等几大类。

（1）国标码。计算机处理汉字所用的编码标准是我国于 1980 年颁布的国家标准 GB 2312－80，即《中华人民共和国国家标准信息交换汉字编码》，简称国标码。国标码的主要用途是作为汉字信息交换码使用。

国标码与 ASCII 码属同一制式，可以认为它是扩展的 ASCII 码。在 7 位 ASCII 码中可以表示 128 个信息，其中字符代码有 94 个。国标码是以 94 个字符代码为基础，其中任何两个代码组成一个汉字交换码，即由两个字节表示一个汉字字符。第一个字节称为“区”，第二个字节称为“位”。这样，该字符集共有 94 个区，每个区有 94 个位，最多可以组成 94×94=8836 个字。

在国标码表中，共收录了一、二级汉字和图形符号 7445 个。其中图形符号 682 个，分布在 1～15 区；一级汉字（常用汉字）3755 个，按汉语拼音字母顺序排列，分布在 16～55 区；二级汉字（不常用汉字）3008 个，按偏旁部首排列，分布在 56～87 区；88 区以后为空白区，以待扩展。

国标码本身也是一种汉字输入码，由区号和位号共 4 位十进制数组成，通常称为区位码输入法。在区位码中，两位区号在高位，两位位号在低位。区位码可以唯一确定一个汉字或字符，反之任何一个汉字或字符都对应唯一的区位码。例如，汉字“啊”的区位码是"1601"，即在 16 区的第 01 位；符号“。”的区位码是"0103"。其"1601"和"0103"是十六进制数。

区位码最大的特点就是没有重码，虽然不是一种常用的输入方式，但对于其他输入方法难以找到的汉字，通过区位码却很容易得到，但需要一张区位码表与之对应。例如，汉字“丰”的区位码是“2365”。

（2）机内码。机内码是指在计算机中表示一个汉字的编码。正是由于机内码的存在，输入汉字时就允许用户根据自己的习惯使用不同的汉字输入码，例如，拼音、五笔、自然、区位等，进入系统后再统一转换成机内码存储。国标码也属于一种机器内部编码，其主要用途是将不同的系统使用的不同编码统一转换成国标码，使不同系统之间的汉字信息进行相互交换。

机内码一般都采用变形的国标码。所谓变形的国标码是国标码的另一种表示形式，即将每个字节的最高位置 1。这种形式避免了国标码与 ASCII 码的二义性，通过最高位来区别是 ASCII 码字符还是汉字字符。

注意：因为汉字的区码和位码的范围都在 01～94 内，所以不直接用区位码作为计算机内码，否则会与基本的 ASCII 码发生冲突。

笔试模拟习题

一、简答题

1．什么是数制？
2．有哪几种常用进制？其特点是什么？
3．在计算机中为什么采用二进制？
4．在计算机中为什么引入八进制和十六进制？
5．什么是信息编码？
6．信息存储的单位有哪些？

7．什么是数值型数据？

8．什么是非数值型数据？

9．机器数的表示方法有几种？分别是什么？

二、计算题

1．将二进制数 100110110111.00101 转换成八进制数、十进制数、十六进制数。

2．将八进制数 604.05 转换成二进制数、十进制数、十六进制数。

3．将十进制数 85.765 转换成二进制数、八进制数、十六进制数。

4．将十六进制数 F05D.7A1 转换为二进制数、八进制数、十进制数。