

§ 1. C++的初步知识

1. 进制的基本概念

★ 进制：按进位原则进行记数的方法叫做**进位记数制**，简称为**进制**

十进制 : 0-9, 逢十进一

二十四进制: 0-23, 逢二十四进一

六十进制 : 0-59, 逢六十进一

=> N进制, $0 \sim N-1$, 逢N进一

★ 基数：指各种位制中允许选用基本数码的个数

十进制 : 0-9 => 基数为10

二十四进制: 0-23 => 基数为24

六十进制 : 0-59 => 基数为60

★ 位权：在N进制数中，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

$$215 = 2 \times 10^2 + 1 \times 10^1 + 5 \times 10^0$$

2的位权是 10^2 (百位, 基数10为底, 序号为2)

1的位权是 10^1 (十位, 基数10为底, 序号为1)

5的位权是 10^0 (个位, 基数10为底, 序号为0)

§ 1. C++的初步知识

2. 二进制的基本概念

- ★ 二进制：基数为2，只有0、1两个数码，逢二进一
- ★ 二进制是计算机内部表示数据的方法，因为计算机就其本身来说是一个电器设备，为了能够快速存储/处理/传递信息，其内部采用了大量电子元件；在这些电子元件中，电路的**通和断**、电压的**高与低**，这两种状态最容易实现，也最稳定、也最容易实现对电路本身的控制。我们将计算机所能表示这样的状态，用0、1来表示、即用二进制数表示计算机内部的所有运算和操作
- ★ 位与字节：每个二进制位只能表示0/1两种状态，当表示较大数据时，所用位数就比较长，为便于管理，每8位称为一个字节 (**位：bit 字节：byte**)
8 bit = 1 byte
bit : 计算机内表示数据的最小单位
byte : 计算机表示数据的基本单位 (数据表示为1-n个byte)

§ 1. C++的初步知识

2. 二进制的基本概念

★ 位与字节：每个二进制位只能表示0/1两种状态，当表示较大数据时，所用位数就比较长，为便于管理，每8位称为一个字节(位：bit 字节：byte)

8 bit = 1 byte

bit : 计算机内表示数据的最小单位

byte : 计算机表示数据的基本单位 (数据表示为1-n个byte)

● 二进制的大数表示及不同单位的换算如表所示

● 简写的时候，b=bit/B=byte

例：某安卓智能手机，存储空间 16GB
某安卓智能电视机，存储空间 32Gb

● 实际表述中，K/M/G 既可以表示 $2^{10}/2^{20}/2^{30}$ ，也可以是 $10^3/10^6/10^9$ ，因此部分表述有二义性，折算时有误差，要根据语境理解
(计算机内一般按二进制理解，其余十进制)

例：某程序猿工资：12K = 12000
宽带速率：20Mbps = 20×10^6 (bit per second)
笔记本内存：8GB = 8×2^{40}
硬盘：1TB = 1×10^{12} (厂商标注)
= 1×2^{40} (计算机理解)

二进制大数的表示单位：

2^{10}	= 1024	=1 KB	(KiloByte)
2^{20}	= 1024 KB	=1 MB	(MegaByte)
2^{30}	= 1024 MB	=1 GB	(GigaByte)
2^{40}	= 1024 GB	=1 TB	(TeraByte)
2^{50}	= 1024 TB	=1 PB	(PeraByte)
2^{60}	= 1024 PB	=1 EB	(ExaByte)
2^{70}	= 1024 EB	=1 ZB	(ZetaByte)
2^{80}	= 1024 ZB	=1 YB	(YottaByte)
2^{90}	= 1024 YB	=1 NB	(NonaByte)
2^{100}	= 1024 NB	=1 DB	(DoggaByte)

§ 1. C++的初步知识

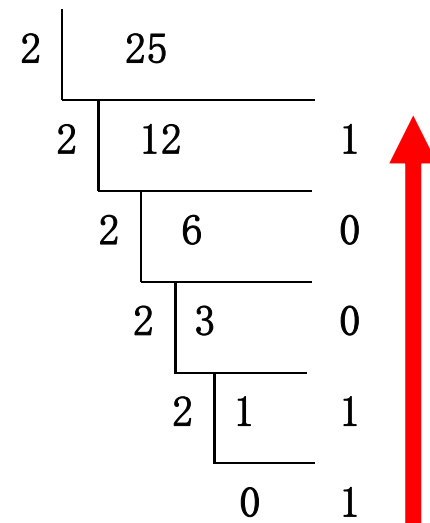
3. 十进制转换为二进制

★ 数制转换：计算机内部采用二进制，但用计算机解决实际问题时，数据的输入输出通常使用十进制(人易于理解)。因此，使用计算机进行数据处理时必须先将十进制转换为二进制，处理完成后再将二进制转换为十进制，这种将数字由一种数制转换成另一种数制的方法称为数制转换

★ 十进制转二进制(整数)：除2取余法

基本方法：

- (1) 要转换的整数除以2，得商和余数(整除)
- (2) 继续用商除以2，得新的商和余数(整除)
- (3) 重复(2)直到商为零时为止
- (4) 把所有余数逆序排列，即为转换的二进制数



★ 二进制转十进制(整数)：按权相加法

基本方法：

- (1) 把二进制数写成加权系数展开式
- (2) 按十进制加法规则求和

$$\begin{aligned} 11001 &= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 16 + 8 + 0 + 0 + 1 \\ &= 25 \end{aligned}$$

★ 暂不考虑负数(第二章讲)

★ 二进制/十进制小数的相互转换(需自学，作业要求)

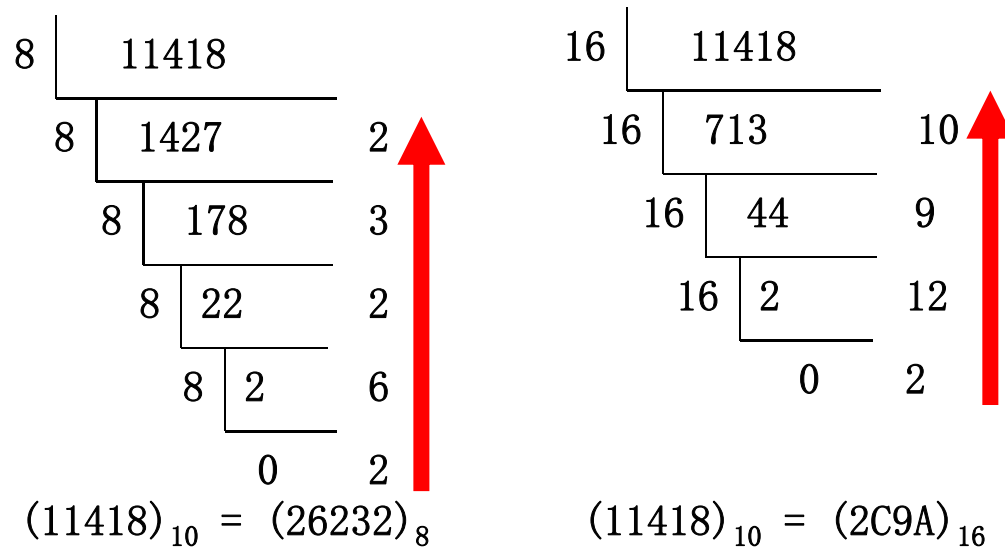
§ 1. C++的初步知识

4. 八进制、十六进制

★ 八进制：基数为8，数码为0-7，逢八进一

★ 十六进制：基数为16，数码为0-15，逢十六进一，为避免歧义，用A-F(a-f)替代10-15表示

★ 十进制转八、十六进制：除8/16取余法



★ 八、十六进制转十进制：按权相加法

$$\begin{aligned}(26232)_8 &= 2 \times 8^4 + 6 \times 8^3 + 2 \times 8^2 + 3 \times 8^1 + 2 \times 8^0 \\ &= 2 \times 4096 + 6 \times 512 + 2 \times 64 + 3 \times 8 + 2 \times 1 \\ &= 11418\end{aligned}$$

$$\begin{aligned}(2C9A)_{16} &= 2 \times 16^3 + 12 \times 16^2 + 9 \times 16^1 + 10 \times 16^0 \\ &= 2 \times 4096 + 12 \times 256 + 9 \times 16 + 10 \times 1 \\ &= 11418\end{aligned}$$

§ 1. C++的初步知识

4. 八进制、十六进制

★ 二进制转八、十六进制：低位开始，每3/4位转1位

$$(1011100101011)_2 = 1\ 011\ 100\ 101\ 011 = (13453)_8$$

$$(1011100101011)_2 = 1\ 0111\ 0010\ 1011 = (172B)_{16}$$

★ 八、十六进制转二进制：每1位转3/4位，不足补0

$$(13453)_8 = 001\ 011\ 100\ 101\ 011 = (1011100101011)_2$$

$$(172B)_{16} = 0001\ 0111\ 0010\ 1011 = (1011100101011)_2$$

★ 暂不考虑负数(第二章讲)

★ 八进制和十六进制的相互转换 (需自学，作业要求)