西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

C程序设计
Programming in C

1011014

主讲：姜学锋，计算机学院

信息在计算机中的表示

1、进制数的转换

1.2.2 进位计数制的转换

- 1．十进制数转换成r进制数
- 由于整数和小数的转换方法不同，将十进制数转换为r进制数时，可分别按整数部分和小数部分转换，然后将结果加起来即可。

1.2.2 进位计数制的转换

- （1）十进制整数转换成r进制数。
- 设有十进制整数I，根据式（1–1）有：

$$I = a_{n-1} \times r^{n-1} + a_{n-2} \times r^{n-2} + \cdots + a_2 \times r^2 + a_1 \times r^1 + a_0 \times r^0$$

式1-2

- 式子两边各除以r，得：

$$\frac{I}{r}=\frac{a_{n-1}\times r^{n-1}+a_{n-2}\times r^{n-2}+\cdots+a_2\times r^2+a_1\times r^1}{r}+\frac{a_0\times r^0}{r}$$

式1-3

- 由于 $a_i<r$ ，因此 $\frac{a_0\times r^0}{r}$ 是I除以r的纯小数或0，即 a_0 是I除以r的余数。

1.2.2 进位计数制的转换

- 显然

$$\frac{a_{n-1}\times r^{n-1}+a_{n-2}\times r^{n-2}+\cdots+a_2\times r^2+a_1\times r^1}{r}=a_{n-1}\times r^{n-2}+a_{n-2}\times r^{n-3}+\cdots+a_2\times r^1+a_1\times r^0$$

- 是I除以r的商，是一个整数。通过这步的计算求出了 a_0。

- 若令I除以r的商为 I'，则：

$$I' = a_{n-1} \times r^{n-2} + a_{n-2} \times r^{n-3} + \cdots + a_2 \times r^1 + a_1 \times r^0$$

式1-4

$$I = a_{n-1} \times r^{n-1} + a_{n-2} \times r^{n-2} + \cdots + a_2 \times r^2 + a_1 \times r^1 + a_0 \times r^0$$

式1-2

- 比较式（1–2）、式（1–4），重复上述步骤可以依次求出 a_0、a_1、$\cdots$ 、a_{n-1} ，即实现了十进制整数转换成r进制数。

- 总结来说，十进制整数转换成r进制数的方法是**除r取余法**：即将十进制整数不断除以r取余数，直到商为0，先得到的余数是 a_0，最后得到的余数是 a_{n-1}，则 $a_{n-1}a_{n-2}\cdots a_1a_0$ 就是转换后的r进制数。

- （2）十进制小数转换成r进制小数。
- 设有十进制小数f，根据式（1–1）有：

$$f = a_{-1} \times r^{-1} + a_{-2} \times r^{-2} + \cdots + a_{-(m-1)} \times r^{-(m-1)} + a_{-m} \times r^{-m}$$

式1-5

1.2.2 进位计数制的转换

- 式子两边各乘以r，得：

$$r \times f = a_{-1} \times r^{0} + a_{-2} \times r^{-1} + \cdots + a_{-(m-1)} \times r^{-(m-2)} + a_{-m} \times r^{-(m-1)}$$

式1-6

- 显然 $a_{-1} \times r^{0}$ 是个整数，因此 a_{-1} 是 $r \times f$ 的整数部分，由此求出了 a_{-1} 。

1.2.2 进位计数制的转换

- 若令 $r \times f - a_{-1}$ 为 f' ，则：

$$f' = a_{-2} \times r^{-1} + \cdots + a_{-(m-1)} \times r^{-(m-2)} + a_{-m} \times r^{-(m-1)}$$

式1-7

$$f = a_{-1} \times r^{-1} + a_{-2} \times r^{-2} + \cdots + a_{-(m-1)} \times r^{-(m-1)} + a_{-m} \times r^{-m}$$

式1-5

- 比较式（1–5）、式（1–7），重复上述步骤可以依次求出 a_{-1}、a_{-2} 、…，即实现了十进制小数转换成r进制小数。

- 需要注意的是， 乘式（1–7）左边的结果的小数部分可能永远不会为0，因此上述步骤可能是无限的。

- 总结来说，十进制小数转换成r进制数的方法是**乘r取整法**：即将十进制小数不断乘以r取整数，直到小数部分为0或达到要求的精度为止，先得到的整数是 a_{-1}，自左向右排列，则 $a_{-1}a_{-2}\cdots$ 就是转换后的r进制小数。

【例1.1】

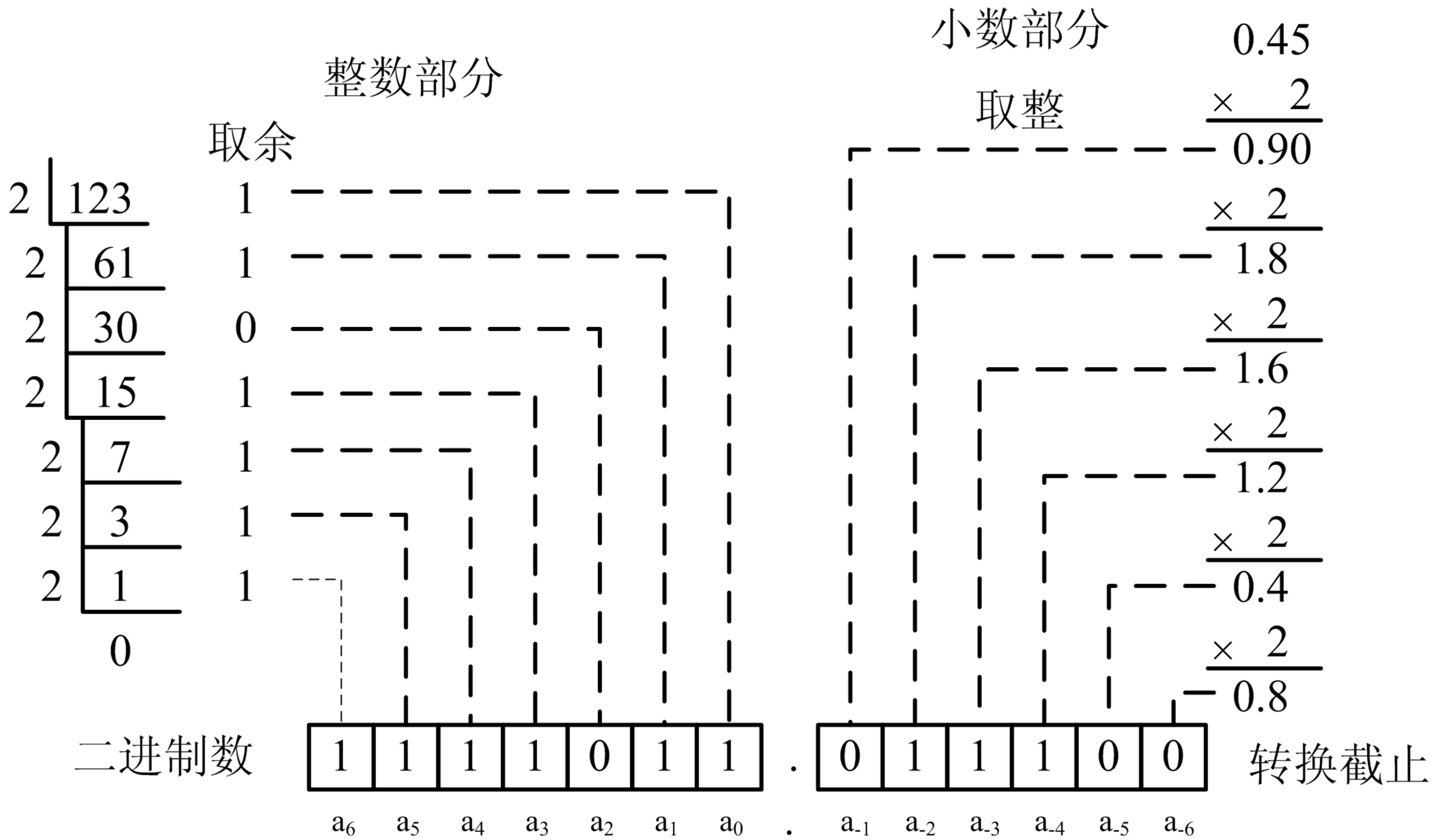
将十进制数 $(123.45)_D$ 转换成二进制数。

解：转换结果为

$$(123.45)_D=(1111011.011100)_B$$

注意，小数部分的转换是不精确的，这里根据精度要求保留6位小数。
转换步骤如下：

1.2.2 进位计数制的转换



【例1.2】

将十进制数 $(12345)_D$ 转换成二进制数。

解：由于转换的十进制数较大，使用除2取余法转换步骤比较多，这里根据二进制位权关系实现快速转换。如图是16位二进制数的位权示意图。

图1.4 二进制位权

2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	2^{-1}	2^{-2}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1	0.5	0.25

因为：$12345 = 8192 + 4096 + 32 + 16 + 8 + 1$ 所以：$(12345)_D = (11000000111001)_B$

【例1.3】

将十进制数 $(123)_D$ 转换成二进制数。

解：由于123靠近 2^7（128），可以使用二进制减法来转换。

即

$$
\begin{array}{r}
10000000 \\
-\quad\quad 101 \\
\hline
1111011
\end{array}
$$

则 $123=128-5=(10000000)_B-(101)_B=(1111011)_B$

- 2．r进制数转换成十进制数
- 将任意r进制数按照式（1–1）写成按位权展开的多项式，各位数码乘以各自的权值且累加起来，就得到该r进制数对应的十进制数。例如：

$$(100101.11)_B=1\times2^5+0\times2^4+0\times2^3+1\times2^2+0\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-2}=(37.75)_D$$

$$(377.65)_O=3\times8^2+7\times8^1+7\times8^0+6\times8^{-1}+5\times8^{-2}=(255.828125)_D$$

$$(7FFF)_H=7\times16^3+15\times16^2+15\times16^1+15\times16^0=(32767)_D$$

- 3．二、八、十六进制数相互转换
- 从前面的例子可以看到，等值的二进制数比十进制数位数要长很多。为了方便起见，在理论分析和程序设计时人们更多使用八进制和十六进制数。

1.2.2 进位计数制的转换

- 二进制、八进制、十六进制之间存在特殊关系：

$$8^1 = 2^3 \quad 16^1 = 2^4$$

- 即1位八进制数相当于3位二进制数，1位十六进制数相当于4位二进制数。根据这种对应关系，可以得到它们之间的转换方法：
- ①二进制数转换成八进制数时，以小数点为中心向左右两边分组，每3位为一组转换成相应的八进制数，两头不足3位用0补；

1.2.2 进位计数制的转换

- ②二进制数转换成十六进制数时，以小数点为中心向左右两边分组，每4位为一组转换成相应的十六进制数，两头不足4位用0补；

- ③八进制数转换成十六进制数或十六进制数转换成八进制数时，可以借助二进制。

1.2.2 进位计数制的转换

▶ 例如：

$$(\underset{3}{\underline{0011}}\ \underset{A}{\underline{1010}}\ \underset{5}{\underline{0101}}\ .\ \underset{D}{\underline{1101}}\ \underset{4}{\underline{0100}})_B = (3A5.D4)_H$$ （整数高位和小数低位补0）

$$(\underset{1}{\underline{001}}\ \underset{6}{\underline{110}}\ \underset{4}{\underline{100}}\ \underset{5}{\underline{101}}\ .\ \underset{6}{\underline{110}}\ \underset{5}{\underline{101}})_B = (1645.65)_O$$ （整数高位补0）

$$(512.E)_H = (\underset{5}{\underline{0101}}\ \underset{1}{\underline{0001}}\ \underset{2}{\underline{0010}}\ .\ \underset{E}{\underline{1110}})_B$$ （整数前的高位0和小数后的低位0可取消）

$$(177.73)_O = (\underset{1}{\underline{001}}\ \underset{7}{\underline{111}}\ \underset{7}{\underline{111}}\ .\ \underset{7}{\underline{111}}\ \underset{3}{\underline{011}})_B$$ （整数前的高位0可取消）

$$(7F)_H = (\underset{7}{\underline{0111}}\ \underset{F}{\underline{1111}})_B = (\underset{1}{\underline{001}}\ \underset{7}{\underline{111}}\ \underset{7}{\underline{111}})_B = (177)_O$$ （借助二进制转换）

CP程序设计