第一章 整数的可除性
2015年03月05日

信息安全数学基础

陈恭亮　教授 博士生导师

上海交通大学信息安全工程学院

chengl@sjtu.edu.cn

1.2 整数的表示

我们平时遇到的整数通常是以10 进制表示. 例如51328 意指

$$5 \cdot 10^4 + 1 \cdot 10^3 + 3 \cdot 10^2 + 2 \cdot 10^1 + 8 \cdot 10^0.$$

中国是世界上最早采用十进制的国家, 春秋战国时期已普遍使用的筹算就严格遵循了十进位制, 见《孙子算经》. 但在计算机中, 51328 要用2 进制, 8 进制或2 进制表示. 为此, 我们考虑一般的b 进制, 再考察特殊的2 进制, 10 进制和16 进制. 运用欧几里得除法, 我们可得到如下定理:

定理1.2.1 设$b > 1$是正整数. 则每个正整数n 可惟一地表示成

$$n = a_{k-1}b^{k-1} + a_{k-2}b^{k-2} + \cdots + a_1 b + a_0,$$

其中a_i是整数,$0 \le a_i \le b-1,\ i = 1, \ldots, k-1$, 且首项系数$a_{k-1} \neq 0$.

存在性

证 先证明n 有表达式. 逐次运用欧几里得除法,
首先, 用b 去除n 得到

$$n = q_0 b + a_0,\ 0 \le a_0 \le b-1.$$

再用b 去除不完全商q_0 得到

$$q_0 = q_1 b + a_1,\ 0 \le a_1 \le b-1.$$

如此可依次得到

$$\begin{aligned} q_1 &= q_2 b + a_2, & 0 \le a_2 \le b-1, \\ &\cdots \\ q_{k-3} &= q_{k-2} b + a_{k-2}, & 0 \le a_{k-2} \le b-1, \\ q_{k-2} &= q_{k-1} b + a_{k-1}, & 0 \le a_{k-1} \le b-1. \end{aligned}$$

因为$0 \le q_{k-1} < \cdots < q_1 < q_0 < n$,
所以必有整数k使得$q_{k-1} = 0$.

惟一性

这样,我们依次得到

$$
\begin{aligned}
n &= q_0 b + a_0 = (q_1 b + a_1) b + a_0 = q_1 b^2 + a_1 b + a_0 \\
&= \dots \\
&= q_{k-3} b^{k-2} + a_{k-3} b^{k-3} + \cdots + a_1 b + a_0 \\
&= q_{k-2} b^{k-1} + a_{k-2} b^{k-2} + \cdots + a_1 b + a_0 \\
&= a_{k-1} b^{k-1} + a_{k-2} b^{k-2} + \cdots + a_1 b + a_0.
\end{aligned}
$$

再证明表示式是惟一的. 如果有两种不同的表示式:

$n = a_{k-1}b^{k-1} + a_{k-2}b^{k-2} + \cdots + a_1 b + a_0,\ 0 \le a_i \le b-1,\ i = 1, \dots, k-1.$

$n = c_{k-1}b^{k-1} + c_{k-2}b^{k-2} + \cdots + c_1 b + c_0,\ 0 \le c_i \le b-1,\ i = 1, \dots, k-1.$

(这里可以取$a_{k-1} = 0$或$c_{k-1} = 0$.) 两式相减得到

$$(a_{k-1} - c_{k-1})b^{k-1} + (a_{k-2} - c_{k-2})b^{k-2} + \cdots + (a_1 - c_1)b + (a_0 - c_0) = 0.$$

假设j 是最小的正整数使得$a_j \neq c_j$, 则

$$b^j((a_{k-1}-c_{k-1})b^{k-1-j}+(a_{k-2}-c_{k-2})b^{k-2-j}+\cdots+(a_{j+1}-c_{j+1})b+(a_j-c_j)) = 0.$$

或者

$$(a_{k-1}-c_{k-1})b^{k-1-j}+(a_{k-2}-c_{k-2})b^{k-2-j}+\cdots+(a_{j+1}-c_{j+1})b+(a_j-c_j) = 0.$$

因此

$$a_j-c_j = -((a_{k-1}-c_{k-1})b^{k-j-2}+(a_{k-2}-c_{k-2})b^{k-j-3}+\cdots+(a_{j+1}-c_{j+1}))b.$$

故

$$b|(a_j-c_j), \quad |a_j-c_j| \geq b.$$

但

$$0 \leq a_j \leq b-1, \quad 0 \leq c_j \leq b-1,$$

又有$|a_j-c_j| < b$. 这不可能. 也就是说n 的表示式是惟一的.

1.2.1 b 进制表示

为了说明整数是关于基b 的表示式, 引进如下符号:

定义1.2.1 我们用$n = (a_{k-1}a_{k-2}\ldots a_1a_0)_b$ 表示展开式:

$$n = a_{k-1}b^{k-1} + a_{k-2}b^{k-2} + \cdots + a_1b + a_0,$$

其中$0 \le a_i \le b-1,\ i = 1, \ldots, k-1,\ a_{k-1} \ne 0$, 并称其为整数$n$ 的**b 进制表示**.

这时, n 的b-进制位数是$k = [\log_b n] + 1$. 事实上,

$$b^{k-1} \le n < b^k \quad \text{或} \quad k-1 \le \log_b n < k.$$

因此, $k - 1 = [\log_b n]$.

例1.2.1 表示整数642 为2 进制.

解 逐次运用欧几里得除法, 我们有

$$\begin{aligned}
642 &= 321\cdot 2 + 0, & 20 &= 10\cdot 2 + 0,\\
321 &= 160\cdot 2 + 1, & 10 &= 5\cdot 2 + 0,\\
160 &= 80\cdot 2 + 0, & 5 &= 2\cdot 2 + 1,\\
80 &= 40\cdot 2 + 0, & 2 &= 1\cdot 2 + 0,\\
40 &= 20\cdot 2 + 0, & 1 &= 0\cdot 2 + 1.
\end{aligned}$$

因此, $642 = (1010000010)_2$, 或者

$$642 = 1\cdot 2^9+0\cdot 2^8+1\cdot 2^7+0\cdot 2^6+0\cdot 2^5+0\cdot 2^4+0\cdot 2^3+0\cdot 2^2+1\cdot 2^1+0\cdot 2^0.$$

各进制间的转换

计算机也常用8 进制, 或16 进制, 或64 进制等. 在16 进制中, 我们用

$$0,\ 1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7,\ 8,\ 9,\ A,\ B,\ C,\ D,\ E,\ F$$

分别表示 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 即

$$A = 10,\ B = 11,\ C = 12,\ D = 13,\ E = 14,\ F = 15.$$

例1.2.2 转换16 进制$(ABC8)_{16}$ 为10 进制.

$$(ABC8)_{16} = 10 \cdot 16^3 + 11 \cdot 16^2 + 12 \cdot 16 + 8 = (43796)_{10}.$$

为了方便各进制转换, 提高效率(以空间换时间), 可预制换算表.

10 进制	16 进制	2 进制	10 进制	16 进制	2 进制
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	1	0011	11	B	1011
4	1	0100	12	C	1100
5	1	0101	13	D	1101
6	1	0110	14	E	1110
7	1	0111	15	F	1111

10 进制	16 进制	2 进制	10 进制	16 进制	2 进制
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	1	0011	11	B	1011
4	1	0100	12	C	1100
5	1	0101	13	D	1101
6	1	0110	14	E	1110
7	1	0111	15	F	1111

例1.2.3 转换16 进制$(ABC8)_{16}$ 为2 进制.

因$A=(1010)_2, B=(1011)_2, C=(1100)_2, 8=(1000)_2$. 从而

$$(ABC8)_{16}=(\underbrace{1010}_{A}\underbrace{1011}_{B}\underbrace{1100}_{C}\underbrace{1000}_{8})_2.$$

例1.2.4 转换2 进制$(10111011111111101001)_2$ 为16 进制.

由上述换算表可得到

$$(1001)_2 = 9,\ (1110)_2 = E,\ (1111)_2 = F,$$

$$(1101)_2 = D,\ (101)_2 = (0101)_2 = 5.$$

从而 $(\underbrace{101}_{5}\underbrace{1101}_{D}\underbrace{1111}_{F}\underbrace{1110}_{E}\underbrace{1001}_{9})_2 = (5DFE9)_{16}.$

因为2 进制的转换比16 进制要容易些, 所以我们可以先将数作2 进制表示, 然后, 运用2 进制与16进制之间的换算表, 将2 进制转换成16 进制.

例1.2.5 表示整数642 为16 进制.

解 根据例1, 我们有 $642 = (1010000010)_2.$

又查换算表得到

$$(0010)_2 = 2, \quad (1000)_2 = 8, \quad (10)_2 = (0010)_2 = 2.$$

故 $642 = 2 \cdot 16^2 + 8 \cdot 16^1 + 2 = (282)_{16}.$

b 进制运算—-加法运算

前面, 我们给出了整数的b 进制表示. 现在讨论的b 进制数的运算.

b**进制加法运算**. 设$n = (a_{k-1} \ldots a_1 a_0)_b,\ m = (b_{k-1} \ldots b_1 b_0)_b$. 则

$$n + m = (c_k c_{k-1} \ldots c_1 c_0)_b$$

的具体运算过程如下(从右到左):

1) 如果$a_0+b_0 < b$, 则取$c_0 = a_0+b_0,\ d_0 = 0$. 否则, 取 $c_0 = a_0+b_0-b,\ d_0 = 1$.

2) 如果$a_1 + b_1 + d_0 < b$, 则取$c_1 = a_1 + b_1 + d_0,\ d_1 = 0$. 否则, 取

$$c_1 = a_1 + b_1 + d_0 - b,\ d_1 = 1. \quad \cdots\cdots$$

k) 如果$a_{k-1} + b_{k-1} + d_{k-2} < b$, 则取$c_{k-1} = a_{k-1} + b_{k-1} + d_{k-2},\ d_{k-1} = 0$. 否则, 取 $c_{k-1} = a_{k-1} + b_{k-1} + d_{k-2} - b,\ d_{k-1} = 1$.

k+1) 最后, 取$c_k = d_{k-1}$.

$$\begin{array}{rc} & a_j + d_{j-1} \\ + & b_j \\ \hline d_j \leftarrow & c_j \end{array} \qquad \begin{array}{rc} & a_j + d_{j-1} \\ + & b_j \\ \hline 0 \leftarrow & a_j + b_j + d_{j-1} \end{array} \qquad \begin{array}{rc} & a_j + d_{j-1} \\ & b_j \\ \hline 1 \leftarrow & a_j + b_j + d_{j-1} - b \end{array}$$

例1.2.6 设$n=(51328)_{10}$, $m=(49138)_{10}$. 则$n+m=(100466)_{10}$.

$$\begin{array}{rrrrrr} & 5 & 1 & 3 & 2 & 8 \\ + & 4 & 9 & 1 & 3 & 8 \\ \hline 1 & 0 & 0 & 4 & 6 & 6 \end{array}$$

b 进制运算—-减法运算

减法运算. 设$n = (a_{k-1}\dots a_1 a_0)_b,\ m = (b_{k-1}\dots b_1 b_0)_b$. 设$n \geq m$. 则

$$a - b = c = (c_{k-1}\dots c_1 c_0)_b$$

的具体运算过程如下(从右到左):

1) 如果$a_0 \geq b_0$, 则取$c_0 = a_0 - b_0,\ d_0 = 0$. 否则, 取 $c_0 = b + a_0 - b_0,\ d_0 = 1$.

2) 如果$a_1 - d_0 \geq b_1$, 则取$c_1 = a_1 - d_0 - b_1,\ d_1 = 0$. 否则, 取

$$c_1 = b + a_1 - d_0 - b_1,\ d_1 = 1. \quad \cdots\cdots$$

k-1) 如果$a_{k-2} - d_{k-3} \geq b_{k-2}$, 则取$c_{k-2} = a_{k-2} - d_{k-3} - b_{k-2},\ d_{k-2} = 0$. 否则, 取 $c_{k-2} = b + a_{k-2} - d_{k-3} - b_{k-2},\ d_{k-2} = 1$.

k) 最后, 取$c_{k-1} = a_{k-1} - d_{k-2} - b_{k-1}$.

$$\begin{array}{ccc} d_j\rightarrow & a_j - d_{j-1} \\ - & b_j \\ \hline & c_j \end{array} \qquad \begin{array}{ccc} 0\rightarrow & a_j - d_{j-1} \\ - & b_j \\ \hline & a_j - d_{j-1} - b_j \end{array} \qquad \begin{array}{ccc} 1\rightarrow & a_j - d_{j-1} \\ - & b_j \\ \hline & b + a_j - d_{j-1} - b_j \end{array}$$

访问主页
标题页
目录页
第 13 页 共 31 页
返回
全屏显示
关闭
退出

例1.2.7

$$
\begin{array}{r}
5\ 1\ 3\ 2\ 8 \\
-\ 4\ 9\ 1\ 3\ 8 \\
\hline
2\ 1\ 9\ 0
\end{array}
$$

b 进制运算—-乘法运算

乘法运算. 设$n = (a_{k-1}a_{k-2}\ldots a_1a_0)_b$, $m = (b_{l-1}b_{l-2}\ldots b_1b_0)_b$. 则

$$n \cdot m = (c_{k+l-1}c_{k+l-2}\ldots c_1c_0)_b$$

的具体运算过程如下(从右到左):

$$n \cdot m = \sum_{i=0}^{k-1} a_i b^i \sum_{j=0}^{l-1} b_j b^j = \sum_{j=0}^{l-1}(\sum_{i=0}^{k-1} a_i b_j b^i) b^j = (c_{k+l-1}c_{k+l-2}\ldots c_1c_0)_b.$$

1) 对$j=0$, 计算 $n \cdot b_0 = (c_{0,k}c_{0,k-1}\ldots c_{0,1}c_{0,0})_b$.

2) 对$j=1$, 计算

$$n \cdot b_1 b + (c_{0,k}c_{0,k-1}\ldots c_{0,1}c_{0,0})_b = (c_{1,k+1}c_{1,k}\ldots c_{1,1}c_{1,0})_b. \quad \ldots\ldots$$

$$\begin{array}{rl}
n \cdot b_{j-1}b^{j-1} \rightarrow & a_{k-1}\cdot b_{j-1}b^{k-1+j-1} + a_{k-2}\cdot b_{j-1}b^{k-2+j-1} + \cdots + a_0 \cdot b_{j-1}b^{0+j-1} \\
+ & c_{j-2,k+j-2}b^{k-1+j-1} + c_{j-2,k+j-3}b^{k-2+j-1} + \cdots c_{j-2,1}b + c_{j-2,0} \\
\hline
c_{j-1,j+l-1}b^{k+j-1} & c_{j-1,k+j-2}b^{k-1+j-1} + c_{j-1,k+j-3}b^{k-2+j-1} + \cdots c_{j-1,1}b + c_{j-1,0}
\end{array}$$

l) 对于$j=l-1$, 计算

$$\begin{aligned} & n\cdot b_{l-1}b^{l-1}+(c_{l-2,k+l-2}c_{l-2,k+l-3}\dots c_{l-2,1}c_{l-2,0})_b \\ = & (c_{l-1,k+l-1}c_{l-1,k+l-2}\dots c_{l-1,1}c_{l-1,0})_b. \end{aligned}$$

l+1) 最后, 取$(c_{k+l-1}c_{k+l-2}\dots c_1c_0)_b=(c_{l-1,k+l-1}c_{l-1,k+l-2}\dots c_{l-1,1}c_{l-1,0})_b$.

例1.2.8 设$n=(51328)_{10},\ m=(128)_{10}$. 则$n\cdot m=(6570184)_{10}$.

$$\begin{array}{ccccccc} & & 5 & 1 & 3 & 2 & 8 \\ & \cdot & & & 1 & 2 & 8 \\ \hline & 4 & 1 & 0 & 6 & 2 & 4 \\ 1 & 0 & 2 & 6 & 5 & 6 & \\ 5 & 1 & 3 & 2 & 8 & & \\ \hline 6 & 5 & 7 & 0 & 1 & 8 & 4 \end{array}$$

b 进制运算—-除法运算

除法运算.设整数$n=(a_{k-1}\dots a_1a_0)_b$, $m=(b_{l-1}\dots b_1b_0)_b$, 不妨设$k\geq l$, $a_{k-1}\neq 0$, $b_{l-1}\neq 0$. 则求商$q=(q_{k-l}\dots q_1q_0)_b$ 和余数$r=(r_t\dots r_1r_0)_b$ 使得

$$n=q\cdot m+r,\quad 0\leq r<m$$

的具体运算过程如下(从右到左):

1) 计算$n_1=(a_{1,k_1-1}a_{1,k_1-2}\dots a_{1,1}a_{1,0})_b=\begin{cases} n-m\cdot b^{k-l}, & 若\ n\geq m\cdot b^{k-l},\\ n-m\cdot b^{k-l-1}, & 若\ n<m\cdot b^{k-l}.\end{cases}$

2) 如果$n_1<m$, 则运算终止. 否则, 计算

$$n_2=(a_{2,k_2-1}a_{2,k_2-2}\dots a_{2,1}a_{2,0})_b=\begin{cases} n_1-m\cdot b^{k_1-l}, & 若\ n_1\geq m\cdot b^{k_1-l},\\ n_1-m\cdot b^{k_1-l-1}, & 若\ n_1<m\cdot b^{k_1-l}.\end{cases}\quad \dots\dots$$

s-1) 如果$n_{s-2}<m$, 则运算终止. 否则, 计算

$$n_{s-1}=(a_{s-1,k_{s-1}-1}\dots a_{s-1,1}a_{s-1,0})_b=\begin{cases} n_{s-2}-m\cdot b^{k_{s-2}-l}, & 若\ n_{s-2}\geq m\cdot b^{k_{s-2}-l},\\ n_{s-2}-m\cdot b^{k_{s-2}-l-1}, & 若\ n_{s-2}<m\cdot b^{k_{s-2}-l}.\end{cases}$$

s) 最后, $n_{s-1}<m$. 我们有$r=n_{s-1}$ 及q 使得 $n=q\cdot m+r,\quad 0\leq r<m.$

b 进制运算—-除法运算

设$n=(a_{k-1}\dots a_1a_0)_b,\ m=(b_{l-1}\dots b_1b_0)_b$, 不妨设$k\geq l$.

$$
\begin{array}{rc}
 & n \\
- & b^{k-l-1}\cdot m \\
\hline
 & n_1
\end{array}
\qquad
\begin{array}{rl}
 & a_{k-1}b^{k-1}+a_{k-2}b^{k-2}+\cdot+a_1b+a_0 \\
- & b_{l-1}b^{k-l-1+l-1}+b_{l-2}b^{k-l-1+l-2}+\cdots+b_0b^{k-l-1+0} \\
\hline
 & a_{1,k_1-1}b^{k_1-1}+a_{1,k_1-2}b^{k_1-2}+\cdot+a_{1,1}b+a_{1,0}
\end{array}
$$

$$
\begin{array}{rc}
 & n_{j-1} \\
- & b^{k_{j-1}-l-1}\cdot m \\
\hline
 & n_j
\end{array}
\qquad
\begin{array}{rl}
 & a_{j-1,k_{j-1}-1}b^{k_{j-1}-1}+a_{j-1,k_{j-1}-2}b^{k_{j-1}-2}+\cdot+a_{j-1,1}b+a_{j-1,0} \\
- & b_{l-1}b^{k_{j-1}-l-1+l-1}+b_{l-2}b^{k-l-1+l-2}+\cdots+b_0b^{k-l-1+0} \\
\hline
 & a_{j,k_j-1}b^{k_j-1}+a_{j,k_j-2}b^{k_j-2}+\cdot+a_{j,1}b+a_{j,0}
\end{array}
$$

$$n_{j-1}-b^{k_{j-1}-l-1}\cdot m=n_j$$

$$(k_{j-1},a_{j-1,k_{j-1}-1})>(k_j,a_{j,k_j-1}).$$

例1.2.9 设$n = (51328)_{10}$, $m = (428)_{10}$. 则$n \cdot m = (119)_{10} \cdot (428)_{10} + (396)_{10}$.

解 由假设$k = 5$, $l = 3$.

i) 因为$n > m \cdot 10^{k-l}$, 所以计算 $n_1 = n - m \cdot 10^{k-l} = (8528)_{10}$.

ii) $k_1 = 4$. 因为$n_1 > m \cdot 10^{k_1-l}$, 所以计算 $n_2 = n_1 - m \cdot 10^{k_1-l} = (4248)_{10}$.

iii) $k_2 = 4$. 因为$n_2 < m \cdot 10^{k_2-l}$, 所以计算 $n_3 = n_2 - m \cdot 10^{k_2-l-1} = (3820)_{10}$.

iv) $k_3 = 4$. 因为$n_3 < m \cdot 10^{k_3-l}$, 所以计算 $n_4 = n_3 - m \cdot 10^{k_3-l-1} = (3392)_{10}$.

v) $k_4 = 4$. 因为$n_4 < m \cdot 10^{k_4-l}$, 所以计算 $n_5 = n_4 - m \cdot 10^{k_4-l-1} = (2964)_{10}$.

vi) $k_5 = 4$. 因为$n_5 < m \cdot 10^{k_5-l}$, 所以计算 $n_6 = n_5 - m \cdot 10^{k_5-l-1} = (2536)_{10}$.

vii) $k_6 = 4$. 因为$n_6 < m \cdot 10^{k_6-l}$, 所以计算 $n_7 = n_6 - m \cdot 10^{k_6-l-1} = (2108)_{10}$.

viii) $k_7 = 4$. 因为$n_7 < m \cdot 10^{k_7-l}$, 所以计算 $n_8 = n_7 - m \cdot 10^{k_7-l-1} = (1680)_{10}$.

ix) $k_8 = 4$. 因为$n_8 < m \cdot 10^{k_8-l}$, 所以计算 $n_9 = n_8 - m \cdot 10^{k_8-l-1} = (1252)_{10}$.

x) $k_9 = 4$. 因为$n_9 < m \cdot 10^{k_9-l}$, 所以计算 $n_{10} = n_9 - m \cdot 10^{k_9-l-1} = (824)_{10}$.

xi) $k_{10} = 3$. 因为$n_{10} > m \cdot 10^{k_{10}-l}$, 所以计算 $n_{11} = n_{10} - m \cdot 10^{k_{10}-l} = (396)_{10}$.

xii) 最后, $n_{11} < m$. 我们有$r = n_{11} = (396)_{10}$, $q = (119)_{10}$ 使得

$$n = q \cdot m + r, \quad 0 \leq r < m.$$

大O 符号和小o 符号

大O 符号和小o 符号. 设$f(n)$ 和$g(n)$ 都是正整数n 的正值函数. 如果存在一个正常数C, 使得对任意的正整数n 都有

$$f(n) \le Cg(n),$$

就称$g(n)$ 是$f(n)$ 的界, 记作 $f(n) = O(g(n))$,
简记为$f = O(g)$. 例如, $f(n) = 2\log_2 n = O(\log_2 n)$.
如果对任意小的正数ϵ, 存在一个正整数N, 使得对任意的正整数$n > N$ 都有

$$f(n) < \epsilon g(n),$$

就称$g(n)$ 是比$f(n)$ 高阶的无穷量, 记作 $f(n) = o(g(n))$,
简记为$f = o(g)$. 例如, $f_1(n) = 2\log_2 n = o(n)$,
$f_2(n) = 2n^2 + 3n + 1 = o(e^n)$.
上述关于单变量的定义可以推广到多变量.

设$f(n_1,\dots,n_k)$ 和$g(n_1,\dots,n_k)$ 都是k-重正整数$n_1,\dots,n_k$ 的正值函数. 如果存在一个正常数C, 使得对任意的k-重正整数$n_1,\dots,n_k$都有

$$f(n_1,\dots,n_k)\le Cg(n_1,\dots,n_k),$$

就称$g(n_1,\dots,n_k)$ 是$f(n_1,\dots,n_k)$ 的界, 记作

$$f(n_1,\dots,n_k)=O(g(n_1,\dots,n_k)),$$

简记为$f=O(g)$. 例如, $f(n,m)=2\log_2 n\log_2 m=O(\log_2 n\log_2 m)$. 如果对任意小的正数$\epsilon$, 存在一个正整数$N$, 使得对任意的正整数$n_i>N,\ 1\le i\le k$ 都有

$$f(n_1,\dots,n_k)<\epsilon g(n_1,\dots,n_k),$$

就称$g(n_1,\dots,n_k)$ 是比$f(n_1,\dots,n_k)$ 高阶的无穷量, 记作$f(n_1,\dots,n_k)=o(g(n_1,\dots,n_k))$, 简记为$f=o(g)$.
例如, $f_1(n,m)=2\log_2 n(\log_2 m)^2=o(nm)$,
$f_2(n,m)=(2n^2+3n+1)m^5=o(e^{nm})$.

1.2.2 计算复杂性

算术运算的时间估计.

在算术运算中, 我们常常需要给出运算的时间估计, 这个估计应该只依赖于算法, 而不依赖于运算工具, 如计算器, 计算机等. 因为计算机运行的是比特运算, 所以我们考虑算术运算所需的比特运算次数.

加法. 设

$$a = (a_{k-1}a_{k-2}\dots a_1a_0)_2,\ b = (b_{k-1}b_{k-2}\dots b_1b_0)_2.$$

则

$$a + b = c = (c_kc_{k-1}\dots c_1c_0)_2$$

的运算为:

$$\begin{array}{cccccc} & a_{k-1} & a_{k-2} & \dots & a_1 & a_0 \\ + & b_{k-1} & b_{k-2} & \dots & b_1 & b_0 \\ \hline c_k & c_{k-1} & c_{k-2} & \dots & c_1 & c_0 \end{array}$$

加法

$$\begin{array}{rcccccc} & a_{k-1} & a_{k-2} & \ldots & a_1 & a_0 \\ + & b_{k-1} & b_{k-2} & \ldots & b_1 & b_0 \\ \hline c_k & c_{k-1} & c_{k-2} & \ldots & c_1 & c_0 \end{array}$$

上述运算的具体过程如下(从右到左):

1) 如果$a_0 + b_0 \leq 1$, 则取$c_0 = a_0 + b_0,\ d_0 = 0$. 否则, 取

$$c_0 = a_0 + b_0 - 2,\ d_0 = 1.$$

2) 如果$a_1 + b_1 + d_0 \leq 1$, 则取$c_1 = a_1 + b_1 + d_0,\ d_1 = 0$. 否则, 取$c_1 = a_1 + b_1 + d_0 - 2,\ d_1 = 1$. ......

k) 如果$a_{k-1}+b_{k-1}+d_{k-2} \leq 1$, 则取$c_{k-1} = a_{k-1}+b_{k-1}+d_{k-2},\ d_{k-1} = 0$. 否则, 取

$$c_{k-1} = a_{k-1} + b_{k-1} + d_{k-2} - 2,\ d_{k-1} = 1.$$

k+1) 最后, 取$c_k = d_{k-1}$.

至多作了$k \leq \max(\log_2 a, \log_2 b) + 1$ 次比特运算.

因此, 时间$(a+b) = O(\max(\log_2 a, \log_2 b))$.

例1.2.10 设$a = 11100110$，$b = 10101010$. 则$a + b = 110010000$.

$$
\begin{array}{r}
1\ 1\ 1\ 0\ 0\ 1\ 1\ 0 \\
+\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0 \\
\hline
1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0
\end{array}
$$

减法

减法. 设$a = (a_{k-1}a_{k-2}\ldots a_1a_0)_2$, $b = (b_{k-1}b_{k-2}\ldots b_1b_0)_2$. 不妨设$a \ge b$. 则$a-b=c=(c_{k-1}\ldots c_1c_0)_2$ 的运算为:

$$\begin{array}{cccccc} & a_{k-1} & a_{k-2} & \ldots & a_1 & a_0 \\ - & b_{k-1} & b_{k-2} & \ldots & b_1 & b_0 \\ \hline & c_{k-1} & c_{k-2} & \ldots & c_1 & c_0 \end{array}$$

上述运算的具体过程如下(从右到左):

1) 如果$a_0 \ge b_0$, 则取$c_0 = a_0 - b_0,\ d_0 = 0$. 否则, 取$c_0 = 2 + a_0 - b_0,\ d_0 = 1$.

2) 如果$a_1 - d_0 \ge b_1$, 则取$c_1 = a_1 - d_0 - b_1,\ d_1 = 0$. 否则, 取

$$c_1 = 2 + a_1 - d_0 - b_1,\ d_1 = 1. \quad \ldots\ldots$$

k-1) 如果$a_{k-2} - d_{k-3} \ge b_{k-2}$, 则取$c_{k-2} = a_{k-2} - d_{k-3} - b_{k-2},\ d_{k-2} = 0$. 否则, 取

$$c_{k-2} = 2 + a_{k-2} - d_{k-3} - b_{k-2},\ d_{k-2} = 1.$$

k) 最后, 取$c_{k-1} = a_{k-1} - d_{k-2} - b_{k-1}$.

至多作了$k \le \max(\log_2 a, \log_2 b) + 1$ 次比特运算.

因此, 时间$(a-b) = O(\max(\log_2 a, \log_2 b))$.

例1.2.11 设$a = 11100110$, $b = 10101010$. 则$a - b = 111100$.

$$\begin{array}{rcccccccc} & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ - & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ \hline & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \end{array}$$

乘法

乘法. 设$a = (a_{k-1}a_{k-2}\ldots a_1a_0)_2$, $b = (b_{l-1}b_{l-2}\ldots b_1b_0)_2$. 不妨设$a \geq b$. 则$a \cdot b = c = (c_kc_{k-1}\ldots c_1c_0)_2$ 的运算为:

$$
\begin{array}{ccccccccc}
 & & & a_{k-1} & \ldots & a_{l-1} & \ldots & a_1 & a_0 \\
 & & \cdot & & & b_{l-1} & \ldots & b_1 & b_0 \\
\hline
 & & & a_{k-1}b_0 & \ldots & a_{l-1}b_0 & \ldots & a_1b_0 & a_0b_0 \\
 & & a_{k-1}b_1 & \ldots & a_{l-1}b_1 & \ldots & a_1b_1 & a_0b_1 & \\
 & & & \ldots & \ldots & \ldots & \ldots & & \\
+ & a_{k-1}b_{l-1} & \ldots & \ldots & a_1b_{l-1} & a_0b_{l-1} & & & \\
\hline
c_{k+l-1} & c_{k+l-2} & \ldots & c_{k-1} & \ldots & c_{l-1} & \ldots & c_1 & c_0
\end{array}
$$

上述运算的具体过程如下(从右到左):
1) 对于j, $0 \leq j \leq l-1$, 分别根据$b_j = 0$ 或1, 我们有
$(a_{k-1}b_j\ a_{k-2}b_j\ \ldots\ a_1b_j\ a_0b_j)_2 = (00\ldots00)_2$ 或 $(a_{k-1}a_{k-2}\ldots a_1a_0)_2$.
2) 从第1行至第l行所进行的加法运算, 至多作了
$(k+1)+(k+2)+\ldots+(k+l-1) = kl + (l-1)l/2 \leq 2kl$ 次比特运算.
因此, 时间$(ab) = O(\log_2 a \log_2 b)$.

例1.2.12 设$a = 11100110,\ b = 10101.$ 则$a \cdot b = 1001011011110.$

$$
\begin{array}{cccccccccccccc}
 & & & & & & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
 & & & & \cdot & & & & & 1 & 0 & 1 & 0 & 1 \\
\hline
 & & & & & & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
 & & & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & & & \\
+ & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & & & & & \\
\hline
 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0
\end{array}
$$

除法

除法. 设$a=(a_{k-1}\dots a_1a_0)_2$, $b=(b_{l-1}\dots b_1b_0)_2$, 不妨设$k\ge l$, $a_{k-1}\ne 0$, $b_{l-1}\ne 0$. 则求商$q=(q_{k-l}\dots q_1q_0)_2$ 和余数$r=(r_t\dots r_1r_0)_2$ 使得

$a=q\cdot b+r,\quad 0\le r<b$ 的具体运算过程如下(从高位到低位):

1) 计算$n_1=(a_{1,k_1-1}a_{1,k_1-2}\dots a_{1,1}a_{1,0})_2=\begin{cases} a-b\cdot 2^{k-l}, & 若\ a\ge b\cdot 2^{k-l},\\ a-b\cdot 2^{k-l-1}, & 若\ a<b\cdot 2^{k-l}.\end{cases}$

2) 如果$n_1<b$, 则运算终止. 否则, 计算

$$n_2=(a_{2,k_2-1}a_{2,k_2-2}\dots a_{2,1}a_{2,0})_2=\begin{cases} n_1-b\cdot 2^{k_1-l}, & 若\ n_1\ge b\cdot 2^{k_1-l},\\ n_1-b\cdot 2^{k_1-l-1}, & 若\ n_1<b\cdot 2^{k_1-l}.\end{cases}\quad\cdots\cdots$$

s-1) 如果$n_{s-2}<b$, 则运算终止. 否则, 计算

$$n_{s-1}=(a_{s-1,k_{s-1}-1}\dots a_{s-1,1}a_{s-1,0})_2=\begin{cases} n_{s-2}-b\cdot 2^{k_{s-2}-l}, & 若\ n_{s-2}\ge b\cdot 2^{k_{s-2}-l},\\ n_{s-2}-b\cdot 2^{k_{s-2}-l-1}, & 若\ n_{s-2}<b\cdot 2^{k_{s-2}-l}.\end{cases}$$

s) 最后, $n_{s-1}<b$. 我们有$r=n_{s-1}$ 及q 使得 $a=q\cdot b+r,\quad 0\le r<b$.

因为$k>k_1>k_2>\cdots>k_{s-1}\ge l$ 及$k_s\ge l$, 所以$s-1\le k-l$. 又每次运算至多作了l 次比特运算. 这样, 总运算次数至多为$(k-l)l$.

因此, 时间$(a=q\cdot b+r)=O((\log_2 a)(\log_2 b))$.

访问主页
标题页
目录页
第 29 页 共 31 页
返回
全屏显示
关闭
退出

例1.2.13 设$a = 111010110,\ b = 10101$. 则$a = q \cdot b + r$.

解 由假设$k = 9,\ l = 5$.

i) 因为$a > b \cdot 2^{k-l}$, 所以计算

$$n_1 = a - b \cdot 2^{k-l} = 10000110.$$

ii) $k_1 = 8$. 因为$n_1 < b \cdot 2^{k_1-l}$, 所以计算

$$n_2 = n_1 - b \cdot 2^{k_1-l-1} = 110010.$$

iii) $k_2 = 6$. 因为$n_2 > b \cdot 2^{k_2-l}$, 所以计算

$$n_3 = n_2 - b \cdot 10^{k_2-l} = 1000.$$

iv) 最后, $n_3 < b$. 我们有$r = n_3 = 1000$, $q = 10110$ 使得

$$a = q \cdot b + r, \quad 0 \leq r < b.$$

典型大数的2-进制长度($k = [\log_2 n] + 1$)

n	k	n	k
天(秒数)	17	巨型计算机10万亿次/秒(10^{12})	40
年(秒数)	25	巨型计算机运算一年	65
万年(秒数)	39	巨型计算机运算万年	79
百万(10^6)	20	$\sqrt{k},\ (k = 2^{512})$	256
亿(10^8)	27	$e^{\sqrt{\log k \log\log k}},\ (k = 2^{512})$	67
行星年龄10^9 年	30	$e^{\sqrt{\log k \log\log k}},\ (k = 2^{1024})$	101
宇宙年龄10^{10} 年	34	$e^{\sqrt{\log k \log\log k}},\ (k = 2^{2048})$	150