数值计算方法与Matlab

一、数值计算的特点

数值计算这门课具有以下几个特点：

(1)数值计算是一门与计算机应用密切结合的实用性很强的学科；

思维方法是归纳法，核心问题是“误差”

(2)数值计算这门课程要讨论连续变量和离散变量两种问题，关心的是数值结果；

(3)数值计算这门课程已成为近代数学的一个重要分支，专门研究数学问题的数值解法。

二、数值计算的内容

- 误差分析
- 非线性方程的数值解法
- 线性方程组的数值解法
- 矩阵分解
- 最小二乘问题的数值解法
- 插值与拟合
- 连续函数的最佳逼近
- 数值积分与数值微分
- 常微分方程的数值解法
- 代数特征值问题*

三 学习重点:

1. **构造数值方法的原理（支撑理论）**

迭代法,以直代曲,化整为零,外推法

2. **评价数值方法的好坏**

(研究数值方法的性态、可靠性、效率)

3. **数值方法的计算机实现（计算机实验）**

本课程的基本目的，通过学习和实验，初步建立并理解数值计算，特别是科学与工程计算的基本概念，为进一步深入的学习打下坚实的基础。

第一章 数值计算与误差分析

第一节　数值算法

第二节　数值计算的误差分析

*第三节　二进制数

*第四节　数学软件工具

第一节 数值算法

算法：从给定的已知量出发，经过有限次四则运算及规定的运算顺序，最后求出未知量的数值解，这样构成的完整计算步骤称为算法。

评价算法的标准：复杂度和精度

时间复杂度：计算量，一个算法所需四则运算总次数；一个算法所需的乘除运算总次数。

空间复杂度：程序运行所需的计算机存储量。

例 计算 x^{255} 两种算法比较。

A: $x^{255} = x \cdot \overbrace{x \cdots x}^{254}$ **254次乘法**

B: $x^{255} = x \cdot x^2 \cdot x^4 \cdot x^8 \cdot x^{16} \cdot x^{32} \cdot x^{64} \cdot x^{128}$ *14次乘法*

- **算法B(Matlab)**

```
s = x;
y = x;
for i = 1:7
    s = s * s;
    y = y * s;
end
```

(输入x, 输出y=x^255)

计算量

$N = 14$ （14次乘法）

存储量=4，s,x,y,i

$$P_n(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$$

算法1、需乘法2n-1次，加法n次，存储单元n+4个。

算法2、Horner算法

$$P_n(x)=x(x(x\cdots(x(a_nx+a_{n-1})+a_{n-2}+\cdots+a_1)\ldots))+a_0$$

有递推公式　　　　需乘法n次，加法n次，存储单元n+3个。

$$\begin{cases} s_n=a_n & \%a_n \\ s_k=xs_{k+1}+a_k & \%a_nx+a_{n-1}\quad (k=n-1,\cdots,2,1,0) \\ P_n(x)=s_0 \end{cases}$$

例 矩阵乘积AB的计算量分析

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \cdots\cdots & \cdots & \cdots & & \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1s} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2s} \\ \cdots\cdots & \cdots & \cdots & & \\ b_{n1} & b_{n2} & b_{n3} & \dots & b_{ns} \end{bmatrix}}_{B} = [c_{ij}]_{m\times s}$$

$$C_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \quad i = 1, \cdots, m; \; j = 1, \cdots, s$$

A B 的计算量为N= (m ×n ×s)flop

第二节 数值计算的误差分析

用计算机解决科学计算问题时，需要经历以下几个环节：

实际问题 → 建立数学模型 → 确定数值计算方法 → 编制程序上机算出结果

实际问题的精确解与用计算机计算出来的数值结果之间就有差异，这种差异在数学上称为误差。

数值结果是指在选择某种数值方法之后，编制程序正确，输入初始数据正确的情形下所获得的数值结果。

一、误差的来源

*模型误差　数学模型与实际问题之间出现的这种误差称为模型误差。

*观测误差　通过仪器观测，确定数学模型中的参数所引起这种的误差称为观测误差。

例　求 e^x 的近似值。

$$e^x = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \cdots + \frac{1}{n!}x^n + \cdots,$$

$$S_3(x) = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3$$

截断误差 模型的准确解与某种数值方法的准确解之间的误差称为截断误差或方法误差。

$$e^x - S_3(x) = \frac{1}{4!}x^4 + \cdots$$ 就是截断误差。

舍入误差 用计算机计算，由于计算机字长有限而在数值运算的每一步所产生的误差称为舍入误差。

用4位浮点机计算 $\frac{1}{6}$ 所产生的误差：

$$\frac{1}{6} - 0.1667 = -0.0000333\cdots$$ 就是舍入误差。

在实数系中，每一个实数可以有无穷位，不同的实数代表数轴上不同的点；

$$\sqrt{3} = 1.732050808\ \cdots$$

在计算机数系中，每一个数只有有限位，只有部分有理数能被计算机数系中的数精确表示。

浮点数：这种允许小数点位置浮动的表示法称为数的浮点形式。

$36.83=0.3683\times10^2=0.03683\times10^3$

实数x的十进制浮点形式为

$$x = \pm\ \underbrace{0.a_1 a_2 \ldots a_k \ldots}_{\text{尾数}} \times \underbrace{10}_{\text{基数}}^{\overbrace{c}^{\text{阶码}}}, \quad (1)$$

$a_i \in \{0, 1, 2, \ldots, 9\}$， $c \in Z$

$a_1 \neq 0$, (1)称为x的规格化的浮点形式

x 的k位规格化十进制机器数

- $y=\pm 0.a_1 a_2 \ldots a_k \times 10^c$, $y=fl(x)$

$a_i \in \{0,1,2,\cdots,9\}$, $a_1 \neq 0$，$L \leq c \leq U$,

k是机器数的字长； L、U 是常数。

x的k位十进制机器数$fl(x)$可用两种方法定义：

(1) 截断式 $x=\pm 0.a_1 a_2 \ldots a_k a_{k+1} \ldots \times 10^c$

$fl(x)=\pm 0.a_1 a_2 \ldots a_k \times 10^c$

(2) 四舍五入式

$$fl(x)=\begin{cases} \pm 0.a_1 a_2 \cdots a_k \times 10^c & a_{k+1}<5 \\ \pm(0.a_1 a_2 \cdots a_k + 10^{-k}) \times 10^c & a_{k+1} \geq 5 \end{cases}$$

一般数制情况： k位规格化机器数

$y=\pm 0.\ \boldsymbol{a}_1\ \boldsymbol{a}_2 \ldots\ \boldsymbol{a}_k \times \beta^c$， $\beta=2, 8, 10, 16,$

$\boldsymbol{a}_i \in \{0, 1, 2, \ldots, \beta-1\}$， $L \le c \le U$, $\boldsymbol{a}_1 \ne 0$

F(β,k.L,U)表示以上数集全体加数0，它是计算机中使用的有限离散数集(机器数系）。

F(β,k,L,U)中的数称为机器数。

F(10,4,-33,33)， $y=\pm 0.\ a_1 a_2 a_3 a_4 \times 10^c$

例 在机器数系 F(10,4,-33,33)中表示 $fl(\pi)$.

$$\pi = 3.1415926\cdots \notin F(10,4,-33,33),$$

但是

$$0.1000 \times 10^{-33} < \pi < 0.9999 \times 10^{33}$$

采用截断式 $fl(\pi)=0.3141\times 10$

采用四舍五入式 $fl(\pi)=0.3142\times 10$

若浮点数的阶码不在[L,U]内，则出现上溢或下溢。

例如 在4位机器数系 F(10,4,-33,33)中输入 0.28×10^{-34} 出现下溢，输入 0.199×10^{35} 出现上溢。

二、截断误差分析

例：（截断误差）

已知 $e^x = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \cdots + \frac{1}{n!}x^n + \cdots$,

求 e^{-1} 的近似值，并估计误差。

解：利用展开式的前三项，取n=2,

$$e^{-1} \approx 1 + (-1) + \frac{1}{2}(-1)^2 = 0.5$$

由Taylor公式：$f(x) = f(x_0) + f'(x_0)(x - x_0) + \cdots +$

$$\frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - x_0)^{n+1}$$

$$R_n(x) = \frac{x^{n+1}}{(n+1)!}e^{\theta x}, \qquad 0 < \theta < 1$$

截断误差 $|R_2| = |e^{-1} - 0.5| \le \frac{1}{3!} < 1.7*10^{-1}$

例： 当$h \to 0$时，有如下Taylor展开式

$$e^h = 1 + h + \frac{1}{2!}h^2 + O(h^3), \sin(h) = h - \frac{1}{3!}h^3 + O(h^5)$$

试确定近似计算公式$e^h + \sin(h) \approx 1 + 2h + \frac{1}{2!}h^2 - \frac{1}{3!}h^3$的截断误差。

解： $$e^h + \sin(h) = 1 + 2h + \frac{1}{2!}h^2 - \frac{1}{3!}h^3 + O(h^3) + O(h^5)$$

$$= 1 + 2h + \frac{1}{2!}h^2 + O(h^3)$$

$e^h + \sin(h)$的截断误差为$O(h^3)$。

三、舍入误差分析

例：舍入误差

$$1.492 \times 1.066 = 1.590472$$

设在一台虚构的4位数字的计算机上计算

$$1.492 \times 1.066 \approx 1.590$$

舍入误差为 0.000472

（程序）：考虑Matlab简单程序

```
format long
x=4/3-1
y=3*x
z=1-y
```

舍入误差对计算结果影响很大

例：考察方程组

$$\begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \frac{11}{6} \\ \frac{13}{12} \\ \frac{47}{60} \end{bmatrix}$$

其解为 $x_1 = x_2 = x_3 = 1$

如果把系数舍入成三位数字

$$\begin{bmatrix} 1.00 & 0.500 & 0.333 \\ 0.500 & 0.333 & 0.250 \\ 0.333 & 0.250 & 0.200 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1.83 \\ 1.08 \\ 0.783 \end{bmatrix}$$

求解得 $x_1 = 1.09,\quad x_2 = 0.488,\quad x_3 = 1.49$

四、绝对误差和相对误差

定义： 设数a是精确值，x是a的一个近似值，记

$e = |a - x|$ 称e为近似值x的绝对误差，

$e_r = \frac{|a - x|}{|a|} = \frac{e}{|a|}$ 称e_r为近似值x的相对误差。

$e = |a - x| \le \delta x$，称δx为数a的近似值x的绝对误差限，

$e_r = \frac{|a - x|}{|a|} \le \delta_r x$，称$\delta_r x$为近似值$x$的相对误差限。

近似值x在允许的误差限ε内，我们就认为是“准确的”。

例　已知准确值a=3.1415926...是一个无限不循环小数，求截取不同位数后的近似值和误差界。

解：截取一位$x_1 = 3$,

$$e_1 = |a - x_1| \approx 0.14 < \frac{1}{2} \times 10^0;$$

截取三位$x_2 = 3.14$,

$$e_2 = |a - x_2| \approx 0.0016 < \frac{1}{2} \times 10^{-2};$$

截取五位$x_3 = 3.1416$,

$$e_3 = |a - x_3| \approx 0.0000074 < \frac{1}{2} \times 10^{-4}。$$

a的近似值的绝对误差都超不过本身末位数的半个单位

是截取相应位数后，所得到的近似数中绝对误差的最小值。

$$x_4 = 3.1415$$

$$e_4 = |a - x_4| \approx 0.000093 < \frac{1}{2} \times 10^{-3}$$

截取五位 $x_3 = 3.1416$,

$$e_3 = |a - x_3| \approx 0.0000074 < \frac{1}{2} \times 10^{-4}。$$

定义 设近似数 $x = \pm 0.a_1 a_2 \cdots a_n \times 10^p$ 的绝对误差限是第 n 位的半个单位，则数 x 有 n 位有效数字。（$a_1 \neq 0, a_i \in \{0, 1, \ldots, 9\}$）

例：设 $x=\sqrt{3}=1.7320508\ \ldots$

x_1=1.73, x_2=1.7321, x_3=1.7320是其近似值,

问它们分别有几位有效数字?

解：$e_1=\left|\sqrt{3}-x_1\right|\approx 0.0021<\dfrac{1}{2}\times 10^{-2}$ 3位

$e_2=\left|\sqrt{3}-x_2\right|\approx 0.00005=\dfrac{1}{2}\times 10^{-4}$ 5位

$e_3=\left|\sqrt{3}-x_3\right|\approx 0.000051<\dfrac{1}{2}\times 10^{-3}$ 4位

当某个量的准确值很小或很大时，相对误差比绝对误差更能反映准确数与近似数的差异。

例:考虑 1. x =11， a=10 ， e=1， e_r=0.1

2. x =1001， a=1000， e=1， e_r=0.001

3. x =1001， a =1101， e=100， e_r=0.1

一个近似值的准确程度，不仅与绝对误差的大小有关，而且与准确值本身的大小有关。

例：设计算机数系为F(10,t,L,U),将实数 $x=\pm 0.a_1a_2\ldots a_ta_{t+1}\ldots\times 10^c,(a_1\neq 0)$,用四舍五入法表为机器数$fl(x)$;求其有效数字、绝对误差限、相对误差限。

解：

$$fl(x)=\pm\bar{a}\times 10^c$$

$$\bar{a}=\begin{cases}0.a_1a_2\cdots a_t, & a_{t+1}<5\\ 0.a_1a_2\cdots a_t+10^{-t}, & a_{t+1}\geq 5\end{cases}$$

绝对误差为：

$$|x-fl(x)|\leq 0.5\times 10^{c-t}$$

$fl(x)$有t位有效数字。

相对误差为

$$e_r=\frac{|x-fl(x)|}{|x|}\leq\frac{0.5\times 10^{c-t}}{0.1\times 10^c}=5\times 10^{-t}$$

注：

（1）在数值计算中，尽可能多地保留近似数的有效数字，有效数字越多， 相对误差越小，计算结果越精确。

（2）对IEEE国际通用标准双精度数系

机器精度 Eps=2.2204e-16

最小实数 Realmin=2.2251e-308

最大实数 Realmin=1.7977e+308

五、数值运算中的误差估计

1、数值运算的绝对误差和相对误差

$u = f(x, y)$　　　f：表示某一种确定的运算

$$f(a,b) = f(x,y) + \frac{\partial f(x,y)}{\partial x}(a-x) + \frac{\partial f(x,y)}{\partial y}(b-y) + \cdots$$

u 的绝对误差

$$e(u) = |u^* - u| = |f(a,b) - f(x,y)|$$

$$\approx \left|\frac{\partial f(x,y)}{\partial x}(a-x) + \frac{\partial f(x,y)}{\partial y}(b-y)\right|$$

$$\le \left|\frac{\partial f(x,y)}{\partial x}\right| e(x) + \left|\frac{\partial f(x,y)}{\partial y}\right| e(y) \le \left|\frac{\partial f}{\partial x}\right| \delta x + \left|\frac{\partial f}{\partial y}\right| \delta y$$

u 的相对误差

$$e_r(u)=\frac{e(u)}{|u|}\approx\left|\frac{\partial f(x,y)}{\partial x}\right|\frac{e(x)}{|u|}+\left|\frac{\partial f(x,y)}{\partial y}\right|\frac{e(y)}{|u|}$$

$$=\left|\frac{\partial f(x,y)}{\partial x}\right|\frac{|x|}{|u|}\frac{e(x)}{|x|}+\left|\frac{\partial f(x,y)}{\partial y}\right|\frac{|y|}{|u|}\frac{e(y)}{|y|}$$

$$=\left|\frac{\partial f(x,y)}{\partial x}\right|\frac{|x|}{|u|}e_r(x)+\left|\frac{\partial f(x,y)}{\partial y}\right|\frac{|y|}{|u|}e_r(y)$$

$$\leq\left|\frac{\partial f}{\partial x}\right|\frac{|x|}{|u|}\delta_r(x)+\left|\frac{\partial f(x,y)}{\partial y}\right|\frac{|y|}{|u|}\delta_r(y)$$

2、和、差、积、商的误差估计式

由以上公式可以推导和 、差、积、商的误差估 计式

(1) $e(x\pm y)\approx e(x)+e(y)$

$$e_r(x\pm y)\approx\left|\frac{x}{x\pm y}\right|e_r(x)+\left|\frac{y}{x\pm y}\right|e_r(y)$$

(2) $e(xy)\approx|y|e(x)+|x|e(y)$

$$e_r(xy)\approx e_r(x)+e_r(y)$$

(3) $e(\frac{x}{y})\approx\left|\frac{y}{y^2}\right|e(x)+\left|\frac{x}{y^2}\right|e(y)$

$$e_r(\frac{x}{y})\approx e_r(x)+e_r(y)$$

3.注意的问题

(1) 防止相近的两数相减

例1:当x较大时,计算 $\sqrt{x+1}-\sqrt{x}$

x=1000,

0.015,807,437,428,958

解: $\sqrt{x+1}-\sqrt{x}=\dfrac{1}{\sqrt{x+1}+\sqrt{x}}$

0.015,807,437,428,956

例2 计算 $\dfrac{1-\cos x}{\sin x}$, $x=1^\circ$

0.008726867790757

解: 当x很小时，分子出现相近数相减，将以上算式变形

0.008726867790759

$$\frac{(1-\cos x)(1+\cos x)}{\sin x(1+\cos x)}=\frac{1-\cos^2 x}{\sin x(1+\cos x)}=\frac{\sin x}{1+\cos x}$$

(2) 防止大数吃小数

当两个绝对值相差很大的数进行加法或减法运算时,绝对值小的数有可能被绝对值大的数"吃掉"从而引起计算结果很不可靠.

例3:在F(10,5,-119,119)中，计算 23456+0.2+0.4+0.4

上式$=0.23456\times10^5+0.000002\times10^5+0.000004\times10^5+0.000004\times10^5=23456$

在上式中，重新排序计算

上式$=0.2+0.4+0.4+23456=1+23456=0.00001\times10^5+0.23456\times10^5=23457$

(3) 防止接近零的数做除数

分母接近零的数会产生溢出错误,因而产生大的误差,此时可以用数学公式化简后再做.

$$y=\frac{1}{\sqrt{1e20+1}-\sqrt{1e20}}=\mathrm{I}nf$$

数据下溢

$$y=\sqrt{1e20+1}+\sqrt{1e20}=2.000000000000000\text{e}+10$$

第二个算式好。

(4) 注意计算步骤的简化,减小运算次数

简化计算步骤是提高程序执行速度的关键，它不仅可以节省时间，还能减少舍入误差。

例4：设A、B、C、D分别是10×20、 20×50、 50×1、 1×100的矩阵，试按不同的算法求矩阵乘积E=ABCD.

解：由矩阵乘法的结合律，可有如下算法

1. E=((AB)C)D. 计算量N=11500 $=10\times20\times50+10\times1\times50+10\times1\times100$

2. E=A(B(CD)). 计算量N=125000

3. E=(A(BC))D. 计算量N=2200

六、算法的数值稳定性

误差的定量分析法：

误差的定性分析法：即研究算法的数值稳定性。

定义：一个算法如果输入数据有扰动（即有误差），而计算过程中舍入误差不增长，则称此算法是数值稳定的，否则称此算法为不稳定的。

例： $y_n = \int_0^1 \frac{x^n}{x+5} dx \qquad n = 0,1,2,\ldots 8$

解:(1)由 $y_n + 5y_{n-1} = \int_0^1 \frac{x^n + 5x^{n-1}}{x+5} dx = \int_0^1 x^{n-1} dx = \frac{1}{n}$

得 y_n 的递推公式

$$\begin{cases} y_0 = \ln 6 - \ln 5 \approx 0.182 = \overline{y}_0 \\ y_n = \frac{1}{n} - 5y_{n-1} \qquad n = 1,\ldots 8 \end{cases}$$

初值 $\overline{y}_0$有误差 $e_0 = y_0 - \overline{y}_0$

$\overline{y}_1$的误差 $e_1 = y_1 - \overline{y}_1 = (1 - 5y_0) - (1 - 5\overline{y}_0)$

$$= -5y_0 + 5\overline{y}_0 = -5e_0$$

同理得 $\bar{y}_n$ 的误差 $e_n = y_n - \bar{y}_n = (-1)^n 5^n e_0$

此算法是不稳定的

$$(2)\because \quad y_n = \int_0^1 \frac{x^n}{x+5} dx$$

$$0 < y_n < y_{n-1} \Rightarrow 5y_{n-1} < y_n + 5y_{n-1} < 6y_{n-1}$$

$$\frac{1}{6n} < y_{n-1} < \frac{1}{5n}$$

$$y_8 \approx \frac{1}{2}(\frac{1}{6\times 9} + \frac{1}{5\times 9}) \approx 0.019$$

由 $y_n + 5y_{n-1} = \frac{1}{n}$

得递推公式

$$\begin{cases} y_8 \approx 0.019 = \bar{\bar{y}}_8 \\ y_{n-1} = \frac{1}{5}\left(\frac{1}{n} - y_n\right) \qquad n = 8,7,\ldots,1 \end{cases}$$

误差 $e_0 = y_0 - \bar{\bar{y}}_0 = -\frac{1}{5}(y_1 - \bar{\bar{y}}_1) = -\frac{1}{5}e_1 = \ldots$

$$= \frac{(-1)^n}{5^n} e_n$$

$\bar{\bar{y}}_0 = 0.18232155$与准确值$\ln 6 - \ln 5$有相同的8位有效数字。此算法是 数值稳定的。

无条件稳定与条件稳定

对任何输入数据都是稳定的算法称为无条件稳定。

对某些数据稳定，而对另一些数据不稳定的算法称为条件稳定。

例：求二次方程 $ax^2+bx+c=0$ 的根。

方法1：$x_{1,2}=\dfrac{-b\pm\sqrt{b^2-4ac}}{2a}$，

当 $b^2>>4ac$ 时算法不稳定。

方法2：$x_1=\dfrac{-b-\mathrm{sign}(b)\sqrt{b^2-4ac}}{2a}$，$x_2=\dfrac{c}{ax_1}$

其中 $\mathrm{sign}(b)=\begin{cases}1 & b\geq 0\\ -1 & b<0\end{cases}$ 此算法无条件稳定。

例 在F(10,4,-19,19)数系中，求解二次方程：

$$x^2 - 320x + 16 = 0$$

解法1　　按求根公式 $x_{1,2} = \dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

解得，x_1=0.3199×10^3，x_2=0.1000 ×10 ;

解法2

$$x_1 = \frac{-b - sign(b)\sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{c}{ax_1} \quad (x_1 x_2 = \frac{c}{a})$$

解得x_1=0.3199 ×10^3

x_2=0.5002 ×10^{-1}

精确解为

x_1=319.950

x_2=0.0500078

$b^2 >> 4ac$

第三节* 二进制数制

- 计算机中使用二进制的原因：
  可行、简易、逻辑性（**0**代表**false**，**1**代表**true**）
- 计算机中使用二进制表示及存储数据。

进位计数制

- 十进制数：由0~9十个数字组成，基数是10，逢十进一
  二进制数：由0和1两个数字组成，基数是2，逢二进一
  八进制数：由0~7八个数字组成，基数是8，逢八进一
  十六进制数：由0~9以及A、B、C、D、E、F十六个数字组成，基数是16，逢十六进一。

- 权值

$(34958)_{10}=3\times10^4+4\times10^3+9\times10^2+5\times10^1+8\times10^0$

10^0、10^1、10^2、10^3、10^4分别为右向左，每一位对应的权值。

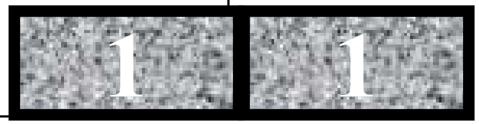
△二进制数与十进制数之间的转换

•十进制整数化成二进制数算法：

除2 取余数，然后倒排余数。

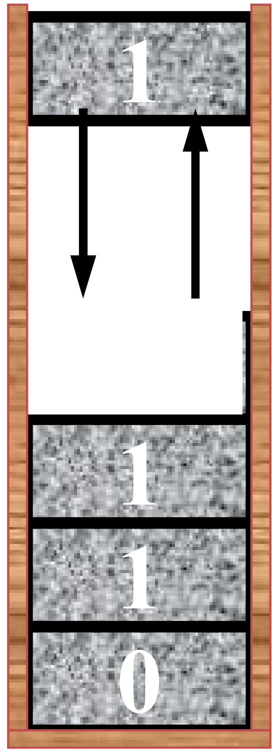
用此方法求：**30 D=（ 1111 0）B**

余数



2	30	…… 0
2	15	…… 1
2	7	…… 1
2	3	…… 1
2	1	…… 1

数字排列方向



除到商为0时止 → 0

△二进制小数与十进制数之间的转换

用乘2取整法将小数部分$(0.6875)_{10}$转换为二进制形式

$$
\begin{array}{r}
0.6875 \\
\times \quad 2 \\
\hline
1.3750
\end{array}
\quad \text{...... 整数部分为1} \quad \text{高位}
$$

$$
\begin{array}{r}
0.3750 \\
\times \quad 2 \\
\hline
0.7500
\end{array}
\quad \text{...... 整数部分为0}
$$

$$
\begin{array}{r}
0.7500 \\
\times \quad 2 \\
\hline
1.5000
\end{array}
\quad \text{...... 整数部分为1}
$$

$$
\begin{array}{r}
0.5000 \\
\times \quad 2 \\
\hline
1.0000
\end{array}
\quad \text{...... 整数部分为1} \quad \text{低位}
$$

$$\boxed{(0.6875)_{10} = (0.1011)_2}$$

不同数制的相互转换

二进制转换为八、十六进制(二、八、十六进制对应关系表如下)

二进制	八进制	十六进制	二进制	八进制	十六进制
000	0	0	1000	10	8
001	1	1	1001	11	9
010	2	2	0101	12	A
011	3	3	1011	13	B
100	4	4	1100	14	C
101	5	5	1101	15	D
110	6	6	1110	16	E
111	7	7	1111	17	F

*第四节 数学软件工具

一、几种常用的数学软件

目前流行的数学软件主要有以下几种：

符号运算软件： Mathematica, Maple

矩阵处理软件： Matlab

统计处理软件： SAS, Spss, Origin

数学CAD软件： MathCAD

第四版习 题

P26-----1, 2, 5