

第一章

1. 计算机图形学的主要研究内容是什么？

答：计算机中图形的表示方法，以及利用计算机进行图形的计算、处理和显示的相关原理与算法，构成了计算机图形学的主要研究内容。图形硬件、图形标准、图形交互技术、光栅图形生成算法、曲线曲面造型、实体造型、真实感图形计算与显示算法，以及科学计算可视化、计算机动画、自然景物仿真、虚拟现实等。

2. 列举三个以上图形学的应用领域。

答：计算机辅助设计与制造（CAD/CAM）、可视化、真实感图形实时绘制与自然景物仿真、计算机动画、用户接口、计算机艺术等。

3. 一个图形系统通常由哪些图形设备组成？

答：一个图形系统通常由图形处理器、图形输入设备和图形输出设备构成。

4. 图形和图像的区别是什么？

答：图像是指计算机内以位图(Bitmap)形式存在的灰度信息；图形含有几何属性，更强调物体（或场景）的几何表示，是由物体（或场景）的几何模型(几何参数)和物理属性(属性参数)共同组成的。

5. CRT 显示器的原理是什么？

答：通过电子枪，将从阴极发射出大量电子，经过强度控制，聚集

和加速，使其形成电子流，再经过偏转线圈的控制，快速的轰击显示器的荧光屏，从而使荧光屏上的荧光粉发亮。

6. LCD 有哪些技术指标？

(1) 可视角度 (2) 点距和分辨率 (3) 展望

7. 有哪些常用的图形输入设备？

答：键盘、鼠标、光笔和触摸屏等。

第二章

1. 描述直线扫描的 DDA 算法、中点画线算法和 Bresenham 算法，并用程序实现 Bresenham 算法。

答：直线扫描的 DDA 算法：

DDA 算法原理是当直线的斜率在 $(-1, 1)$ 之间时，它的增量应该以 x 为准，即每次都应该让 x 增加一个像素，然后计算出 y 的值（ y 值增量绝对值小于 1，更精确），再进行四舍五入，最终得到实际点的像素。当直线斜率绝对值大于 1 时，增量应该以 y 为准。

中点画线算法：假定直线斜率 k 在 $0 \sim 1$ 之间，当前像素点为 (x_p, y_p) ，则下一个像素点有两种可选择点 $P_1(x_p+1, y_p)$ 或 $P_2(x_p+1, y_p+1)$ 。若 P_1 与 P_2 的中点 $(x_p+1, y_p+0.5)$ 称为 M ， Q 为

理想直线与 $x=x_p+1$ 垂线的交点。当 M 在 Q 的下方时，则取 P2 应为下一个像素点；当 M 在 Q 的上方时，则取 P1 为下一个像素点。

Bresenham 算法： 过各行、各列像素中心构造一组虚拟网格线，按直线从起点到终点的顺序计算直线各垂直网格线的交点，然后确定该列像素中与此交点最近的像素。

2. 用中点画线法扫描转换从点 A (1, 0) 到 B (4, 7) 经过的直线段，并给出每一步的判别值。

解：由于 A B 直线的斜率 $k=7/3$ 且 $k>1$

我们将坐标进行交换，得到的 $k=3/7$

就可以用书上的方法：

$a=-3$ $b=7$ $d=1$ $d1=-6$ $d2=8$

判别式随坐标的变化：

x	y	d
0	1	1
1	1	-5
2	2	3
3	2	-3
4	3	5

5 3 -1

6 4 7

7 4 1

程序: `x=1:1:4;`

`y=7/3*x-7/3;`

`a=[1 1 2 2 3 3 4 4];`

`b=0:7;`

`plot (x, y, 'k', a, b, '*r')`

`title(' MATLAB 图形')`

`xlabel(' x=1:4')`

`ylabel(' y=0:7')`

`grid`

3. 描述多边形扫描转换的扫描线算法，并写出伪码。

P26-27

4. 字符串裁剪可按哪三个精度进行？

答：串精度，字符精度，笔画或像素精度。

5. 为了在显示器等输出设备上输出字符，系统中必须装备有相应的字

库。字库中存储了每个字符的形状信息，字库分为哪两种类型？各有什么特点？

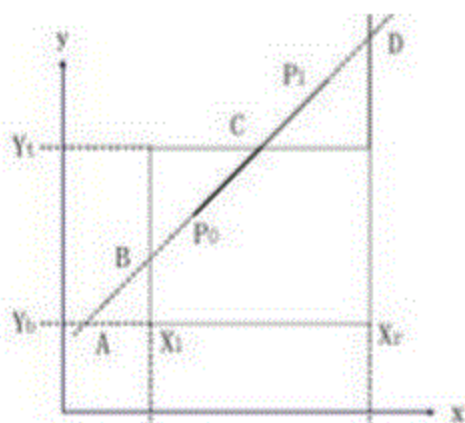
答：字库分为矢量型和点阵型两种。P32

6. 简述裁剪方法和中点裁剪方法的思想，并指出中点裁剪方法的改进之处及这种改进的理由。

答：（1）裁剪就是确定图形中哪些部分落在显示区之内，哪些落在显示区之外，最后只需显示落在显示区内的那部分图形，以便提高显示效率的过程。一般的裁剪方法是：先裁剪再扫描转换。（2）中点裁剪方法的思想是首先对线段端点进行编码，并把线段与窗口的关系分为三种情况，即在全在窗口内、完全不在窗口内和线段与窗口有交。对第一种情况，显示该线段；对第二种情况，丢弃该线段；对第三种情况，用中点分割法求出线段与窗口的交点，即从线段的一端的端点出发找出距该端点最近的可见点，并从线段的另一端点出发找出距该端点最近的可见点，两个可见点之间的连线即为线段的可见部分。（3）中点裁剪方法改进之处：对第三种情况，不直接解方程组求交，而是采用二分法搜索交点。这样改进的理由是：计算机屏幕的像素通常为 1024×1024 ，最多十次二分搜索即可到像素级，必然能找到交点，而且中点法的主要计算过程只用到加法和除 2 运算，效率高，也适合硬件实现。

7. 是描述 Liang-Barskey 裁剪算法，并说明在什么情况下它比中点法和 Cohen-Sutherland 快及原因。

答：（1）设要裁剪的线段是 P_0P_1 。 P_0P_1 和窗口边界交于 A, B, C, D 四点，见图。算法的基本思想是从 A, B 和 P_0 三点中找出最靠近 P_1 点，图中要找的点是 P_0 。从 C, D 和 P_1 中找出最靠近 P_0 的点。图中要找的点是 C 点。那么 P_0C 就是 P_0P_1 线段上的可见部分。（2）梁友栋-Barsky 算法只能应用于矩形窗口的情形。通常梁友栋-Barsky 算法比 Cohen-Sutherland 算法效率更高，因为需要计算的交点数目减少了。更新参数 u_1 、 u_2 仅仅需要一次除法；线段与窗口边界的交点仅计算一次，就计算出 u_1 、 u_2 最后的值。相比之下，即使一条线段完全落在裁剪窗口之外，Cohen-Sutherland 算法也要对它反复求交点，而且每次求交计算都需要做乘除法。



8. 解释走样及反走样的概念，并描述反走样的主要方法。

答：用离散量表示连续量引起的失真现象称为走样，用于减少或消

除这种效果的技术称为反走样。反走样的主要方法有三种：（1）提高分辨率，把显示器分辨率提高一倍，直线经过两倍的像素，锯齿也增加一倍，但同时每个阶梯也减少了一倍，所以显示出的直线看起来就平直光滑了一些。增加分辨率虽然简单，但不是经济的方法，而且它也只能减轻而不能消除锯齿问题。（2）区域采样，区域采样方法假定每个像素是一个具有一定面积的小区域，将直线段看做具有一定宽度的狭长矩形。当直线段与像素有交时，求出两者相交区域的面积，然后根据相交区域面积的大小确定该像素的亮度值。（3）加权区域采样，使相交区域对像素亮度的贡献依赖于该区域与像素中心的距离。当直线经过该像素时，该像素的亮度 F 是在两者相交区域 A 上对滤波器进行积分的积分值。

9. 描述消隐的扫描线 Z-Buffer 算法，并与其他两种 Z-Buffer 算法进行比较。

答：（1）算法的思想描述：在处理当前扫描线时，开一个一维数组作为当前扫描线的 Z-Buffer。首先找出与当前扫描线相关的多边形，以及每个多边形中相关的边对；然后计算每个边对之间的小区间上各像素的深度，并与 Z-Buffer 中的值相比较，找出各像素处对应的可见平面，计算颜色，写帧缓存。对深度计算，采用增量算法。

（2）与 Z-Buffer 算法相比，扫描线 Z-Buffer 算法做了如下两点的改进。

一、将整个绘图窗口内的消隐问题分解到一条条扫描线上解决，使所需的 Z 缓存器大大减少。

2、 计算深度值时，利用了面的连贯性，只用了一个加法。但它在每个像素处都计算深度值，进行 深度比较。因此，被多个多边形覆盖的像素区处还要进行多处计算，计算量仍然很大。

10. 比较书中举例的几种消隐算法的优缺点

答：常见的面消隐算法有：画家算法、Z 缓冲区 (Z-Buffer) 算法、扫描线 Z-buffer 算法、扫描线算法、区域子分割算法 、光线投射算法等。

Z 缓冲区算法是一种典型的、也是最简单的图象空间的消隐算法。

另一个优点是算法便于硬件实现，并可以并行化。

缺点：1) 需要一个额外的 Z 缓冲器 2) 在每个多边形占据的每个像素处都要计算深度值，计算量大 3) 没有利用图形的相关性与连续性。 画家算法它的缺点是只能处理互不相交的面. 扫描线算法和 Z—buffer 算法的缺点是，对于不可见的多边形面片了同样画出，这样造成了绘制过程中不必要的费时。BSP 树算法利用它的存储结构可以优化多边形的排序过程，故它的排序速度比画家算法要快，尤其是复杂度高的场景。

第三章

1. 参数曲线曲面有几种表示形式？

(1) 代数形式 一条三次曲线的代数形式是：

$$\begin{cases} x(t) = a_{3x}t^3 + a_{2x}t^2 + a_{1x}t + a_{0x} \\ y(t) = a_{3y}t^3 + a_{2y}t^2 + a_{1y}t + a_{0y} \\ z(t) = a_{3z}t^3 + a_{2z}t^2 + a_{1z}t + a_{0z} \end{cases} \quad t \in [0,1]$$

(2) 几何形式

描述参数曲线的条件有：端点位矢、端点切矢、曲率等。

$$P(t) = F_0 P_0 + F_1 P_1 + G_0 P'_0 + G_1 P'_1 \quad t \in [0,1]$$

上式是三次 Hermite (Ferguson) 曲线的几何形式， F_0 , F_1 , G_0 , G_1 称为调和函数（或混合函数）

有参数和非参数（显式、隐式）表示

2. 设有控制顶点为 $P_0(0, 0)$, $P_1(48, 96)$, $P_2(120, 120)$, $P_3(216, 72)$ 的三次 Bézier 曲线 $P(t)$ ，试计算 $P(0.4)$ 的 (x, y) 坐标，并写出 $(x(t),$

$y(t)$ 的多项式表示。

$$\begin{aligned}\because P(t) &= \sum_{i=0}^3 P_i B_{i,3}(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t)P_2 + t^3 P_3 \\ \therefore P(0.4) &= (0.6)^3 P_0 + 1.2(0.6)^2 P_1 + 1.8(0.4)^2 P_2 + (0.4)^3 P_3 \\ &= 0.216 \begin{bmatrix} 0 & 0 \end{bmatrix} + 0.432 \begin{bmatrix} 48 & 96 \end{bmatrix} + 0.288 \begin{bmatrix} 120 & 120 \end{bmatrix} + 0.064 \begin{bmatrix} 216 & 72 \end{bmatrix} \\ &= \begin{bmatrix} 69.12 & 80.64 \end{bmatrix} \\ \begin{cases} x(t) = (1-t)^3 x_0 + 3t(1-t)^2 x_1 + 3t^2(1-t)x_2 + t^3 x_3 \\ y(t) = (1-t)^3 y_0 + 3t(1-t)^2 y_1 + 3t^2(1-t)y_2 + t^3 y_3 \end{cases}\end{aligned}$$

3. 设一条二次 Bezier 曲线的控制顶点为 P_0 、 P_1 和 P_2 ， 另一条二次 Bezier 曲线的顶点是 Q_0 、 Q_1 和 Q_2 ， 写出两条曲线精确合并成一条二次 Bezier 曲线的条件。

解：如下图所示，由于可以精确合并，说明两曲线是由一条曲线在参数 $0 < t < 1$ 如下图所示，假设原曲线的控制顶点为 P_0, X, Q_2 。由 de Casteljau 算法，有：

1. 首先要求 $P_1, P_2(Q_0), Q_1$ 三点共线

$$2. \frac{Q_2 - Q_1}{Q_1 - X} = \frac{Q_1 - Q_0}{P_2 - P_1} = \frac{X - P_1}{P_1 - P_0},$$

$$\text{于是有：} Q_1 - \frac{Q_2 - Q_1}{Q_1 - Q_0}(P_2 - P_1) = P_1 + \frac{P_1 - P_0}{P_2 - P_1}(Q_1 - Q_0)$$

4. 已知 Bezier 曲线上的 4 个点分别为 $Q_0(50, 0)$ 、 $Q_1(100, 0)$ 、 $Q_2(0, 50)$ 和 $Q_3(0, 100)$ ，它们对应的参数分别为 $0, 1/3, 2/3, 1$ ，反求 Bezier 曲线的控制点。

答：由已知条件得 $Q_0(50, 0)$, $Q_1(100, 0)$, $Q_2(0, 50)$, $Q_3(0, 100)$

$$Q_0 = P_0 \quad (t=0)$$

$$Q_3 = P_3 \quad (t=1)$$

由公式 $C(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3$

$$Q_1 = (8/27)P_0 + (4/9)P_1 + (2/9)P_2 + (1/27)P_3 \quad (t=1/3)$$

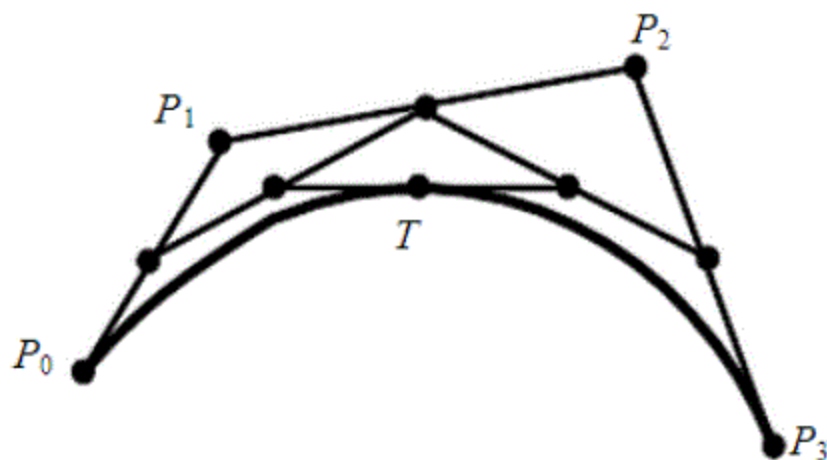
$$Q_2 = (1/27)P_0 + (2/9)P_1 + (4/9)P_2 + (8/27)P_3 \quad (t=2/3)$$

将 Q_0, Q_1, Q_2, Q_3 的 X, Y 坐标代入方程组, 得

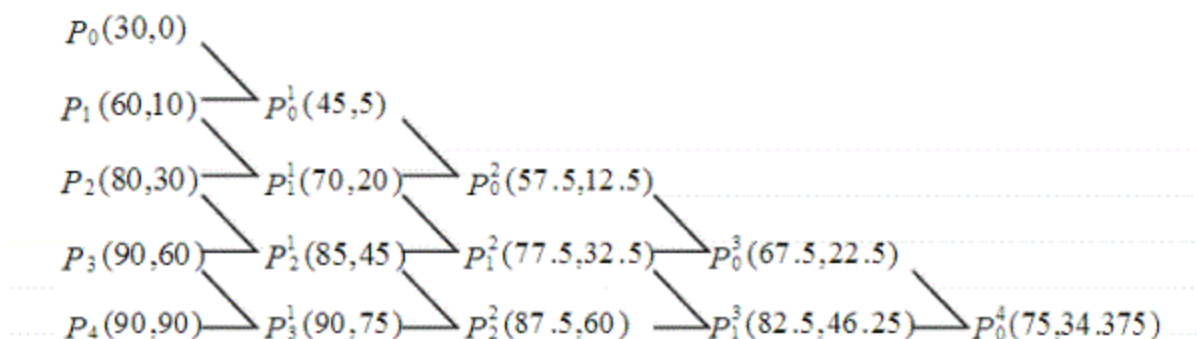
$$X_0 Y_0 (50, 0) \quad X_1 Y_1 (37, 37) \quad X_2 Y_2 (37, 37) \quad X_3 Y_3 (0, 100)$$

5. 设一条三次 Bézier 曲线的控制顶点为 P_0, P_1, P_2, P_3 。对曲线上一点 $P(0.5)$, 及一个给定的目标点 T , 给出一种调整 Bézier 曲线形状的方法, 使得 $P(0.5)$ 精确通过点 T 。

根据 Bézier 曲线的递推算法, 构造过程:



6. 计算以 $(30, 0)$, $(60, 10)$, $(80, 30)$, $(90, 60)$, $(90, 90)$ 为控制顶点的 4 次 Bézier 曲线在 $t=1/2$ 处的值, 并画出 de Casteljau 三角形。



7. 给定三次 Beizer 曲线的控制顶点 $(1, 0)$, $(1, 0, 100)$,
 $(100, 0)$, $(100, 100)$, 计算升阶一次后的控制顶点。

解:

设给定原始控制顶点 P_0, P_1, \dots, P_n , 定义了一条 n 次 Bezier 曲线:

$$P(t) = \sum_{i=0}^n P_i B_{i,n}(t) \quad t \in [0,1]$$

增加一个顶点后, 仍定义同一条曲线的新控制顶点为 $P_0^*, P_1^*, \dots, P_{n+1}^*$, 则有:

$$\sum_{i=0}^n C_n^i P_i t^i (1-t)^{n-i} = \sum_{i=0}^{n+1} C_{n+1}^i P_i^* t^i (1-t)^{n-i}$$

对上式左边乘以 $(t + (1-t))$ ，得到：

$$\sum_{i=0}^n C_n^i P_i t^i (1-t)^{n+1-i} + t^{i+1} (1-t)^{n-i} = \sum C_{n+1}^i P_i^* t^i (1-t)^{n+1-i}$$

比较等式两边 $t^i (1-t)^{n+1-i}$ 项的系数，得到：

$$P_i^* C_{n+1}^i = P_i C_n^i + P_{n-1} C_n^{i-1}$$

化简即得：

$$P_i^* = \frac{i}{n+1} P_{i-1} + \left(1 - \frac{i}{n+1}\right) P_i \quad (i = 0, 1, \dots, n+1)$$

其中 $P_{-1} = P_{n+1} = 0$ 。

升阶一次后的控制顶点为 $(0,0)$ ， $(0,75)$ ， $(50,50)$ ， $(100,25)$ ， $(100,100)$ 。

8. 用 de Boor 算法，求以 $(30, 0)$ ， $(60, 10)$ ， $(80, 30)$ ， $(90, 60)$ ， $(90, 90)$ 为控制顶点，以 $T=[0, 0, 0, 0, 0.5, 1, 1, 1, 1]$ 为节点向量的三次 B 样条曲线在 $t=1/4$ 处的值。

$\because k=4, n=4, k-1 \leq j \leq n$ 即 $3 \leq j \leq 4$

$\therefore 5$ 个控制顶点控制两段三次 B 样条曲线，分别在区间 $[t_3, t_4)$ 和 $[t_4, t_5)$

$\because t_3 \leq t=1/4 \leq t_4$

$\therefore P(t=1/4)$ 在第一段三次 B 样条曲线上， $t \in [t_3, t_4)$ ，该段曲线只与前四个顶点相关

由 de Boor 递推公式

$$P_i^{[r]}(t) = \begin{cases} P_i, r=0, i=j-k+1, j-k+2, \dots, j \\ \frac{t-t_i}{t_{i+k-r}-t_i} P_i^{[r-1]}(t) + \frac{t_{i+k-r}-t}{t_{i+k-r}-t_{i-1}} P_{i-1}^{[r-1]}(t), \\ r=1, 2, \dots, k-1; i=j-k+r+1, j-k+r+2, \dots, j \end{cases}$$

及 $T=[0, 0, 0, 0, 0.5, 1, 1, 1, 1]$ ，可得：

$$P_1^{[1]} = \frac{t-t_1}{t_4-t_1} P_1 + \frac{t_4-t}{t_4-t_1} P_0 = 2tP_1 + 2\left(\frac{1}{2}-t\right)P_0 = \frac{1}{2}P_1 + \frac{1}{2}P_0 = (45,5)$$

$$P_2^{[1]} = \frac{t-t_2}{t_5-t_2} P_2 + \frac{t_5-t}{t_5-t_2} P_1 = tP_2 + (1-t)P_1 = \frac{1}{4}P_2 + \frac{3}{4}P_1 = (35,15)$$

$$P_3^{[1]} = \frac{t-t_3}{t_6-t_3} P_3 + \frac{t_6-t}{t_6-t_3} P_2 = tP_3 + (1-t)P_2 = \frac{1}{4}P_3 + \frac{3}{4}P_2 = (82.5,37.5)$$

$$P_2^{[2]} = \frac{t-t_2}{t_4-t_2} P_2^{[1]} + \frac{t_4-t}{t_4-t_2} P_1^{[1]} = 2tP_2^{[1]} + 2\left(\frac{1}{2}-t\right)P_1^{[1]} = \frac{1}{2}P_2^{[1]} + \frac{1}{2}P_1^{[1]} = (40,10)$$

$$P_3^{[2]} = \frac{t-t_3}{t_5-t_3} P_3^{[1]} + \frac{t_5-t}{t_5-t_3} P_2^{[1]} = tP_3^{[1]} + (1-t)P_2^{[1]} = \frac{1}{4}P_3^{[1]} + \frac{3}{4}P_2^{[1]} \\ = (46.875,20.625)$$

$$P_3^{[3]} = \frac{t-t_3}{t_4-t_3} P_3^{[2]} + \frac{t_4-t}{t_4-t_3} P_2^{[2]} = 2tP_3^{[2]} + 2\left(\frac{1}{2}-t\right)P_2^{[2]} = \frac{1}{2}P_3^{[2]} + \frac{1}{2}P_2^{[2]} \\ = (43.4375,15.3125) = P\left(\frac{1}{2}\right)$$

9. 试证明 n 次 Bezier 曲线退化为 n-1 次 Bezier 曲线的条件为 $\Delta^n P_0 = 0$ 。

证明：n 次 Bezier 曲线退化为 n-1 次 Bezier 曲线的条件为最高次项的系数为 0，由 Bezier 曲线的定义， $P(t) = P_0 C_n^0 t^0 (1-t)^n + P_1 C_n^1 t^1 (1-t)^{n-1} + \dots + P_n C_n^n t^n (1-t)^0$ ，最高项的系数恰好为 P_0 的 n 阶差分，故 n 次 Bezier 曲线退化为 n-1 次 Bezier 曲线的条件为 $\Delta^n P_0 = 0$ 。

10. NURBS 曲线的凸包性指什么？

答：NURBS 曲线的凸包性指的是定义在非零节点区间 $t \in [t_i, t_{i+1}] \subset [t_{k-1}, t_{n+1}]$ 上曲线段位于定义它的 $k+1$ 个控制顶点 P_{i-k+1}, \dots, P_i 的凸包内。整条 NURBS 曲线位于所有定义各曲线段的控制顶点的凸包的并集内。所有权因子的非负性，保证了凸包性质的成立。

11. Q, Q1, Q2, S1, S2 是平面上的 5 个点。请设计一条均匀三次 B 样条曲线，使曲线经过这 5 个点，且满足如下设计要求：

- (1) 在 Q1, Q2 点与 Q 相切；
- (2) 分别在 Q, Q1 和 Q, Q2 间生成一段直线段；
- (3) 在 Q 是一尖点。

答：首先了解均匀三次 B 样条曲线的端点性质。

对于每一段曲线，

已知： $k=4, n=3, T=[0,1,2,3,4,5,6,7]$

所以： $k-1 \leq j \leq n$ 即 $j=3, t \in [t_3, t_4)$

起点： $t=3$

$$\begin{aligned} P(3) &= P_3^{[3]} = (t-3)P_3^{[2]} + (4-t)P_2^{[2]} = P_2^{[2]} = \frac{t-2}{2}P_2^{[1]} + \frac{4-t}{2}P_1^{[1]} \\ &= \frac{1}{2}P_2^{[1]} + \frac{1}{2}P_1^{[1]} = \frac{1}{2}\left(\frac{1}{3}P_2 + \frac{2}{3}P_1\right) + \frac{1}{2}\left(\frac{2}{3}P_1 + \frac{1}{3}P_0\right) = \frac{1}{6}P_0 + \frac{2}{3}P_1 + \frac{1}{6}P_2 \end{aligned}$$

确交线的近似结果。

4) 跟踪方法。跟踪方法求交是通过先求出初始交点，然后从已知的初始交点出发，相继跟踪计算出下一交点，从而求出整条交线的方法

13. 用几何法求平面和球的交线。

答：可按以下步骤求解：

(1) 求球心到平面的距离，设为 d ，交点（投影点）为 P ；

(2) 设球的半径为 r ，若 $r < d$ ，则平面与球相离；

(3) 若 $r = d$ ，则平面与球相切，切点为 P ；

(4) 若 $r > d$ ，则平面与球相交，交线为圆，圆心为 P ，半径为

$$\sqrt{r^2 - d^2}。$$

14. 形体表示有哪些常见的方法？

P94

15. 网格简化时如何度量删除一个顶点的误差？

P118