图形学作业

Chapter 2

2.1

中点画线算法：

思想：在画直线段的过程中，假设当前像素点为(x, y)，下一个像素点的选择有$\begin{cases}(x+1, y)\\(x+1, y+1)\end{cases}$两种可能，此时需要判断理想直线与横坐标 $x+1$ 的交点 Q 离上面哪个点更近，具体做法是构造判别式 $F(x, y) = ax + by + c$，并将中点$(x+1, y+0.5)$代入判别式，以此类推，代码如下：

```
void MidpointLine(int x0, int y0, int x1, int y1, int color) {
    int a, b, d1, d2, d, x, y;
    a = y0 - y1, b = x1 - x0, d = 2 * a + b; // 直线其实是由(x0, y0), (x1, y1)两点确定的，点斜式->一般式
    d1 = 2 * a, d2 = 2 * (a + b);
    x = x0, y = y0; // 初始值
    drawpixel(x, y, color);
    while (x < x1) {
        if (d < 0) {
            // 中点在 Q 下方，取(x + 1, y + 1)
            x += 1, y += 1, d += d2;
        }
        else {
            x += 1, d += d1;
        }
        drawpixel(x, y, color);
    }
}
```

Bresenham 算法：类似中点法，由误差项符号决定下一个像素取右边点还是右上点，代码如下：

```
void BresenhamLine (int x0, int y0, int x1, int y1, int color) {
    int x, y, dx, dy, i, e;
    dx = x1 - x0, dy = y1 - y0, e = -dx;
    x = x0, y = y0;
    for (i = 0; i <= dx; i++) {
        drawpixel (x, y, color);
        x++, e = e + 2 * dy;
```

```
            if (e >= 0)   {
                y++, e = e - 2 * dx;
            }
        }
    }
```

2.2

$(1,0) \to (4,7) \to y = \frac{7-0}{4-1}(x-1) = \frac{7}{3}(x-1)$

$\Rightarrow \underline{k>1} \Rightarrow$ 将x, y互换

直线L: $x = \frac{3}{7}(y-1)$ $\underline{(0,1) \to (7,4)}$

$\Rightarrow 3x - 7y + 7 = 0$

$a=-3, b=7$

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

将(x,y)互换 $\longrightarrow$ 得到原始答案

$(1,0) \to (1,1) \to (2,2) \to (2,3) \to (3,4) \to (3,5) \to (4,6) \to (4,7)$

2.4

字符串裁剪精度：串精度，字符精度，像素精度

2.5

字库的两种类型：点阵型，矢量性

点阵型：每个字符由位图表示，存储空间大，一般由压缩技术解决问题

矢量型：记录字符的笔画信息，存储空间小，美观，变换方便

2.8

走样：用离散量表示连续量引起的失真现象

反走样：减少/消除走样效果的技术，主要方法：提高分辨率，区域采样，加权区域采样，半色调技术

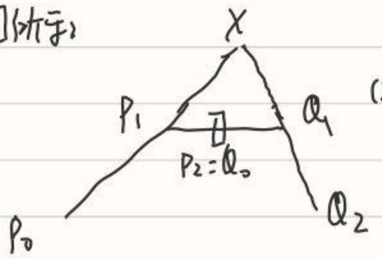
Chapter 3

3.3

3.3.

解：由于可以精确合并 ∴ 两曲线是由一条曲线在参数 $0<t<1$ 处分割而来

如下图所示：



要求：(1) $P_1, P_2(Q_0), Q_1$ 三点共线

(2) $$\frac{Q_2-Q_1}{Q_1-X} = \frac{Q_1-Q_0}{P_2-P_1} = \frac{X-P_1}{P_1-P_0}$$

消除 X 可得：

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

3.4

3.4.

解、设控制顶点分别为 $P_0(Q_0), P_1, P_2, P_3(Q_3)$.

将参数 $1/3$, $2/3$ 分别代入：

$$\begin{cases} \frac{8}{27}P_0+\frac{4}{9}P_1+\frac{2}{9}P_2+\frac{1}{27}P_3=(100,0) \\ \frac{1}{27}P_0+\frac{2}{9}P_1+\frac{4}{9}P_2+\frac{8}{27}P_3=(0,50) \end{cases}$$

$$\Rightarrow \begin{cases} \frac{8}{27}\times 50+\frac{4}{9}x_1+\frac{2}{9}x_2=100 \\ \frac{1}{27}\times 50+\frac{2}{9}x_1+\frac{4}{9}x_2=0 \end{cases} \Rightarrow \begin{cases} x_1=\frac{775}{3} \\ x_2=-\frac{400}{3} \end{cases}$$

$$\begin{cases} \frac{4}{9}y_1+\frac{2}{9}y_2+\frac{1}{27}\times 100=0 \\ \frac{2}{9}y_1+\frac{4}{9}y_2+\frac{8}{27}\times 100=50 \end{cases} \Rightarrow \begin{cases} y_1=-\frac{125}{3} \\ y_2=\frac{200}{3} \end{cases}$$

$$\Rightarrow P_1\left(\frac{775}{3}, -\frac{125}{3}\right) \quad P_2\left(-\frac{400}{3}, \frac{200}{3}\right)$$

3.5

3.5.

解：原先：$P_0, P_1, P_2, P_3 \xrightarrow{\text{调整}} P_0, P_1+\lambda, P_2, P_3$，使 $P(\frac{1}{2})$ 精确通过 T

令改变的曲线为 $P'(t)$

$$P'(t) = P_0 B_{0,3}(t) + (P_1+\lambda) B_{1,3}(t) + P_2 B_{2,3}(t) + P_3 B_{3,3}(t)$$

$$= P(t) + \lambda B_{1,3}(t)$$，将 $t=\frac{1}{2}$ 代入

$$\Rightarrow T = P(\tfrac{1}{2}) + \lambda B_{1,3}(\tfrac{1}{2}) \quad \therefore \lambda = \frac{T - P(\frac{1}{2})}{B_{1,3}(\frac{1}{2})} \Rightarrow \text{将 } P_1 \to P_1 + \frac{T - P(\frac{1}{2})}{B_{1,3}(\frac{1}{2})}$$

3.6

3.6. 递推公式: $p_i^k = \begin{cases} P_i, & k=0 \\ (1-t)p_i^{k-1} + t\,p_{i+1}^{k-1}, & k=1,2,\dots,n,\ i=0,1,\dots,n-k \end{cases}$

$P_0\ (30,0)$

$P_1\ (60,10) \xrightarrow{t} p_0^1\ (45,5)$ 　$(1-t)$

$P_2\ (80,30) \to p_1^1\ (70,20) \to p_0^2\ (57.5,12.5)$

$P_3\ (90,60) \to p_2^1\ (85,45) \to p_1^2\ (77.5,32.5) \to p_0^3\ (67.5,22.5)$

$P_4\ (90,90) \to p_3^1\ (90,75) \to p_2^2\ (87.5,60) \to p_1^3\ (82.5,46.25) \to p_0^4\ (75,34.375)$

3.7

3.7. 解: 升阶公式 $P_i^* = \frac{i}{n+1}P_{i-1} + \left(1-\frac{i}{n+1}\right)P_i\ (i=0,1,\dots,n+1)$, 其中 $P_{-1}=P_{n+1}=(0,0)$

次数

Bi次

$\Rightarrow\ P_0^* = P_0 = (0,0)$

$P_1^* = \frac{1}{4}P_0 + \frac{3}{4}P_1 = \frac{1}{4}(0,0) + \frac{3}{4}(0,100) = (0,75)$

$P_2^* = \frac{1}{2}P_1 + \frac{1}{2}P_2 = \frac{1}{2}(0,100) + \frac{1}{2}(100,0) = (50,50)$

$P_3^* = \frac{3}{4}P_2 + \frac{1}{4}P_3 = \frac{3}{4}(100,0) + \frac{1}{4}(100,100) = (100,25)$

$P_4^* = P_3 = (100,100)$

3.9

3.9、n次Bezier展开式如下：

$P(t) = P_0 B_{0,n}(t) + \cdots + P_n B_{n,n}(t)$

$= \sum_{i=0}^{n} P_i C_n^i t^i (1-t)^{n-i}$，最高项$t^n$系数为$\sum_{i=0}^{n} (-1)^i C_n^{n-i} P_{n-i}$

而$\Delta^k P_t = \Delta(\Delta^{k-1} P_t) = \Delta^{k-1} P_{t+1} - \Delta^{k-1} P_t = \sum_{i=0}^{k} (-1)^i C_k^i P_{t+k-i}$

$\therefore \Delta^n P_0 = \sum_{i=0}^{n} (-1)^i C_n^i P_{n-i} = t^n$系数

又$\because$ n次Bezier曲线退化到n-1次$\Leftrightarrow t^n$系数$=0 \Leftrightarrow \Delta^n P_0 = 0$

3.14

形体表示：分解表示，构造表示，边界表示