计算机图形学

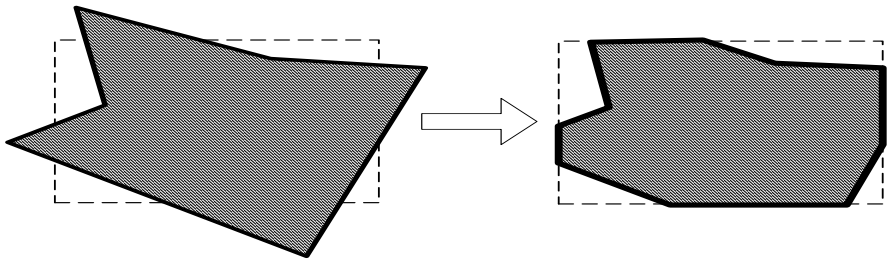
第二章：光栅图形学算法

光栅图形学算法的研究内容

- 直线段的扫描转换算法
- 多边形的扫描转换与区域填充算法
- 直线裁剪算法
- 反走样算法
- 消隐算法

一、裁剪

使用计算机处理图形信息时，计算机内部存储的图形往往比较大，而屏幕显示的只是图形的一部分。



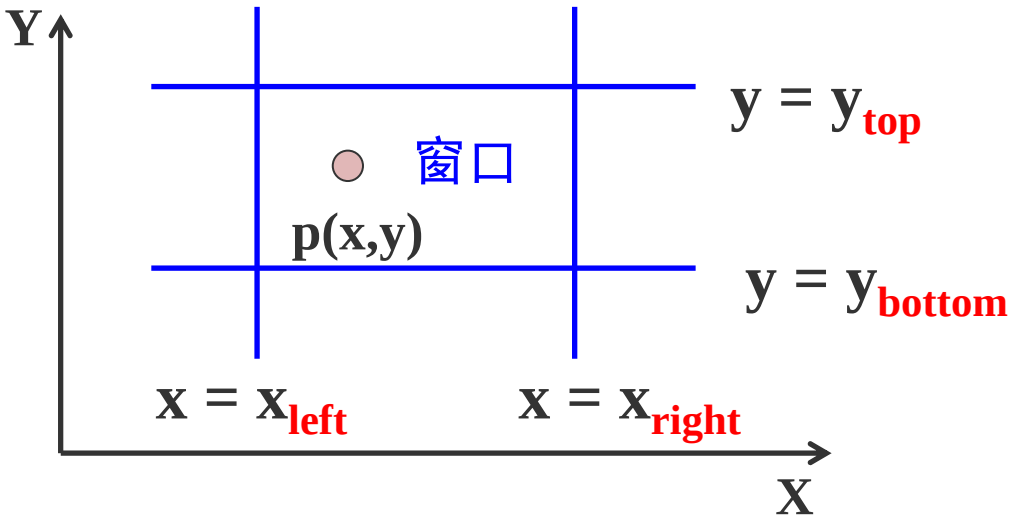
因此需要确定图形哪些部分落在显示区之内，哪些落在显示区之外。这个选择的过程就称为**裁剪**。

最简单的裁剪方法是把各种图形扫描转换为点之后，再判断点是否在窗口内。

1、点的裁剪

对于任意一点P（x，y），
若满足下列两对不等式：

$$\begin{cases} x_{left} \le x \le x_{right} \\ y_{bottom} \le y \le y_{top} \end{cases}$$



则点P在矩形窗口内；否则，点P在矩形窗口之外

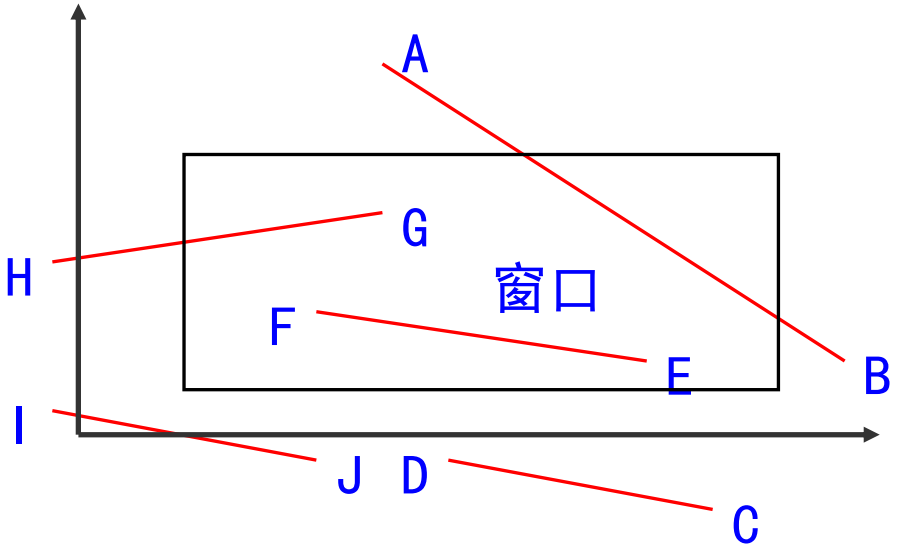
判断图形中每个点是否在窗口内，太费时，一般不可取

2、直线段的裁剪

直线段裁剪算法复杂图形裁剪的基础

直线段和剪裁窗口的可能关系：

- 完全落在窗口内
- 完全落在窗口外
- 与窗口边界相交

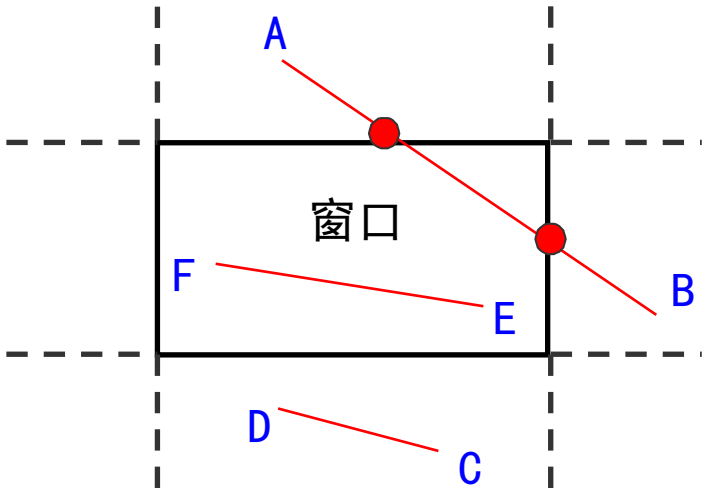


要裁剪一条直线段，首先要判断：

（1）它是否完全落在裁剪窗口内？

（2）它是否完全在窗口外？

（3）如果不满足以上两个条件，则计算它与一个或多个裁剪边界的交点

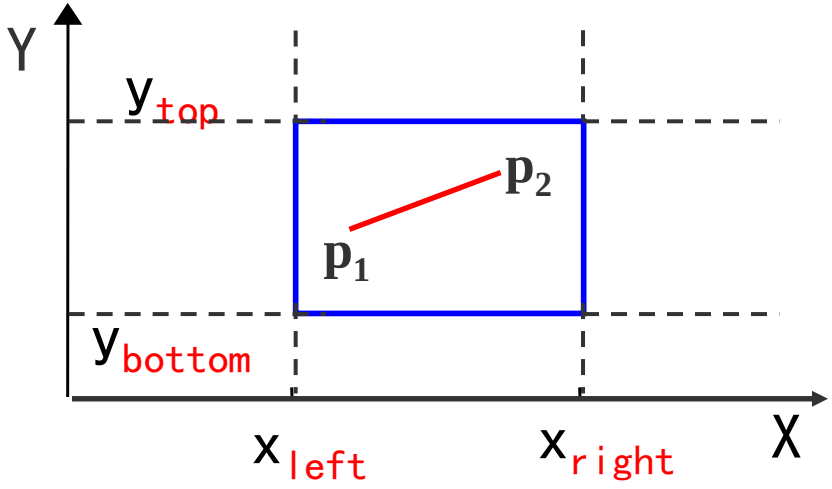


常用的裁剪算法有三种，即Cohen-Sutherland、中点分割法和Liang-Barsky裁剪算法

1、Cohen-Sutherland算法

本算法又称为编码裁剪算法，算法的基本思想是对每条直线段分三种情况处理：

（1）若点p_1和p_2完全在裁剪窗口内



“简取”之

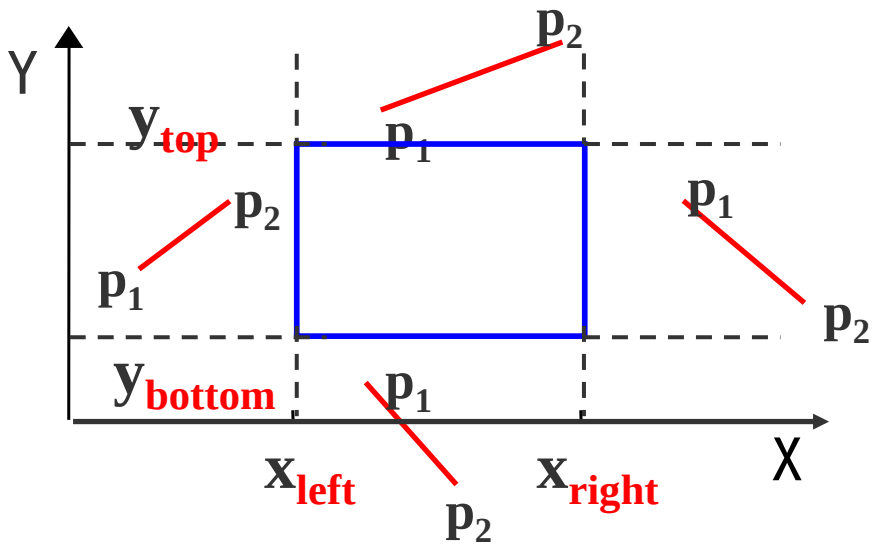
(2) 若点$p_1(x_1, y_1)$和$p_2(x_2, y_2)$均在窗口外，且满足下列四个条件之一：

$x_1 < x_{left}$ 且 $x_2 < x_{left}$

$x_1 > x_{right}$ 且 $x_2 > x_{right}$

$y_1 < y_{bottom}$ 且 $y_2 < y_{bottom}$

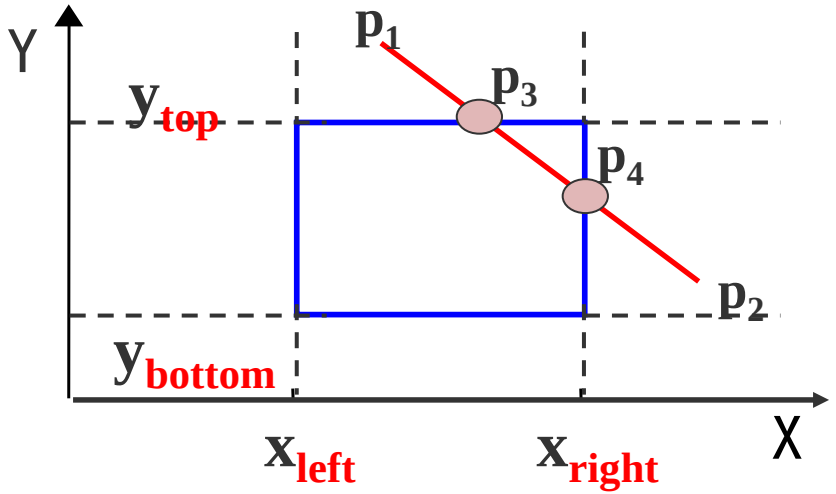
$y_1 > y_{top}$ 且 $y_2 > y_{top}$



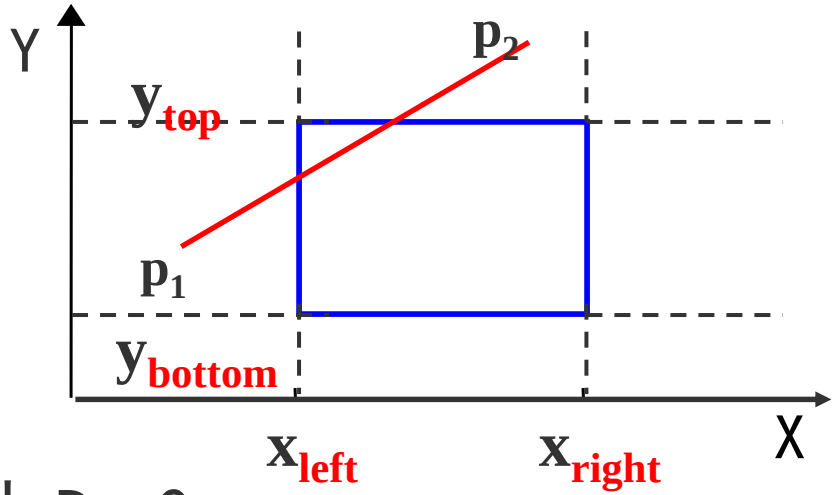
对这四种类型的直线，“简弃”之

（3）如果直线段既不满足“简取”的条件，也不满足“简弃”的条件？

需要对直线段按交点进行分段，分段后判断直线是“简取”还是“简弃”。



每条线段的端点都赋以四位二进制码$D_3D_2D_1D_0$，编码规则如下：



- 若$x<x_{left}$，　则 $D_0=1$，　否则 $D_0=0$
- 若$x>x_{right}$，则 $D_1=1$，　否则 $D_1=0$
- 若$y<y_{bottom}$，则 $D_2=1$，　否则 $D_2=0$
- 若$y>y_{top}$，　则 $D_3=1$，　否则 $D_3=0$

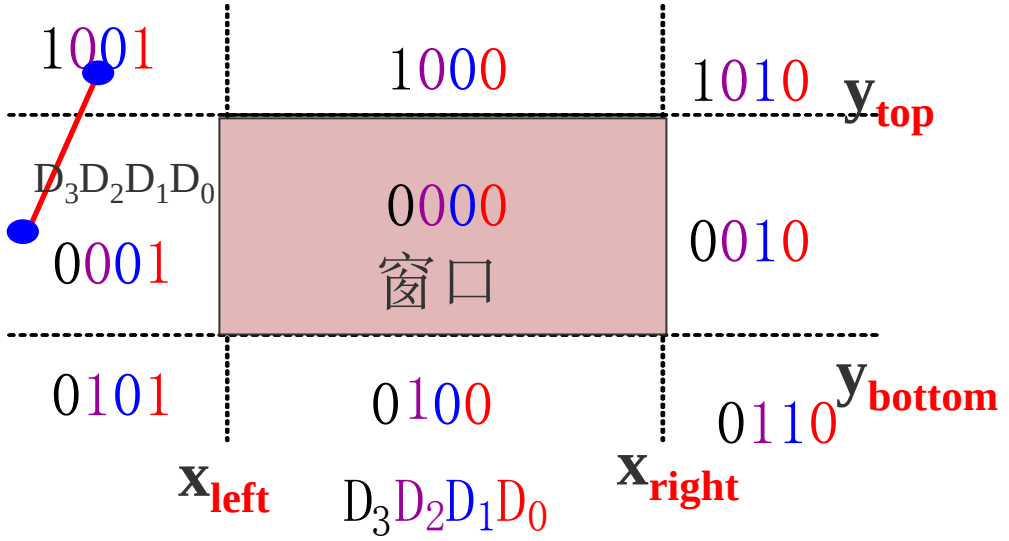
窗口及其延长线所构成了9个区域。根据该编码规则：

D_0对应窗口左边界

D_1对应窗口右边界

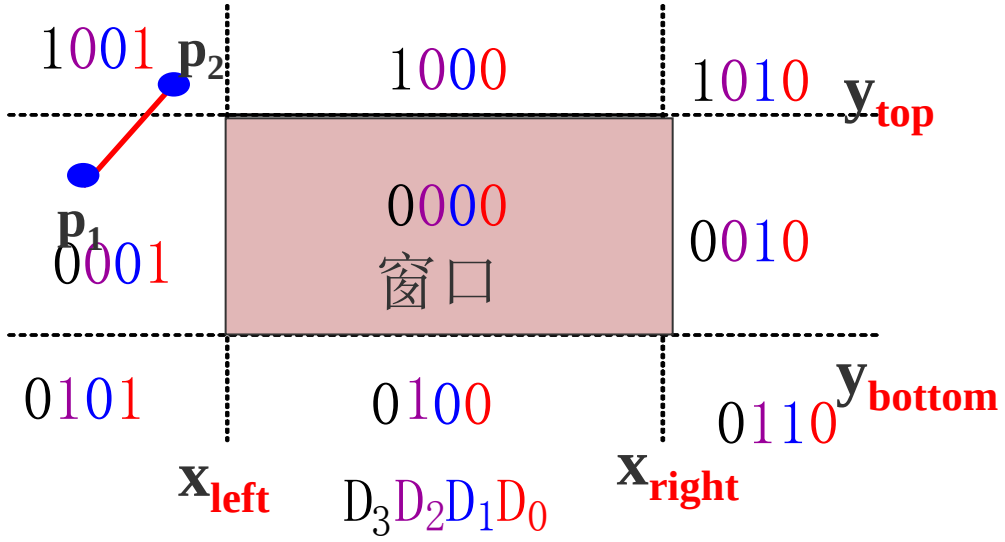
D_2对应窗口下边界

D_3对应窗口上边界



裁剪一条线段时，先求出端点p_1和p_2的编码$code_1$和$code_2$

然后进行二进制“或”运算和“与”运算

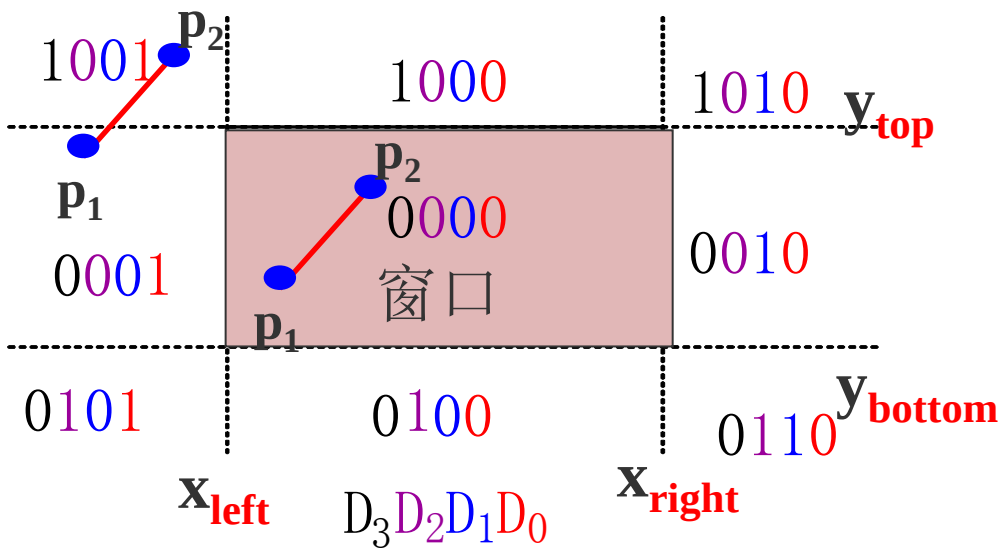


二进制运算

运算符	名称	例子	运算功能
~	位反	~b	求b的位反
&	与运算	b&c	b和c位与
|	或运算	b|c	b和c位或
^	异或运算	b^c	B和c位异或

（1）若$code_1 | code_2=0$，对直线段应简取之

或
0000
0000
0000



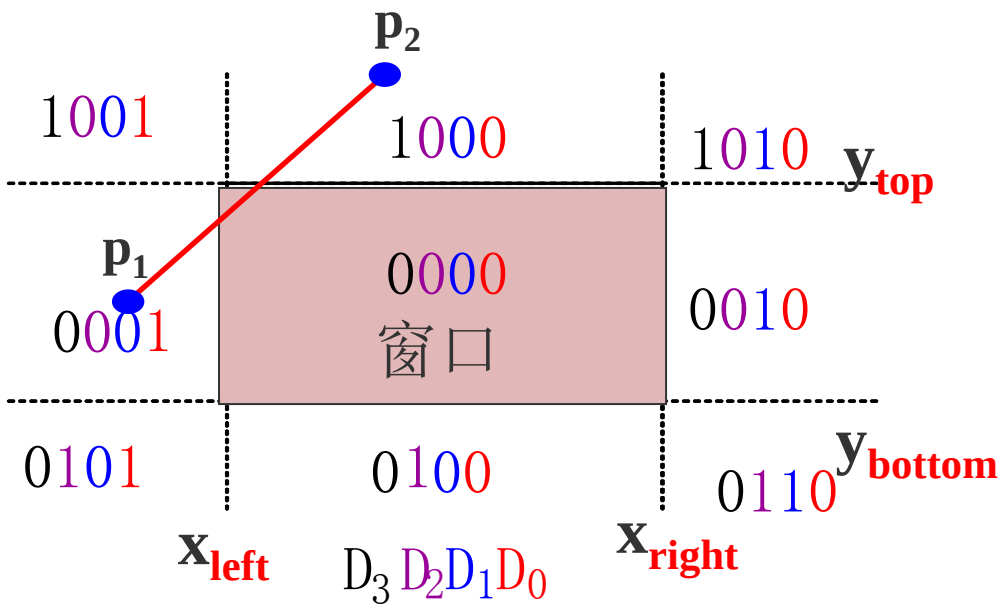
（2）若$code_1 \& code_2 \neq 0$，对直线段可简弃之

与
1001
0001
0001

若上述两条件均不成立

或 0001
1000
1001

与 0001
1000
0000



则需求出直线段与窗口边界的交点在交点处把线段一分为二

下面根据该算法步骤来裁剪如图所示的直线段P_1P_2：

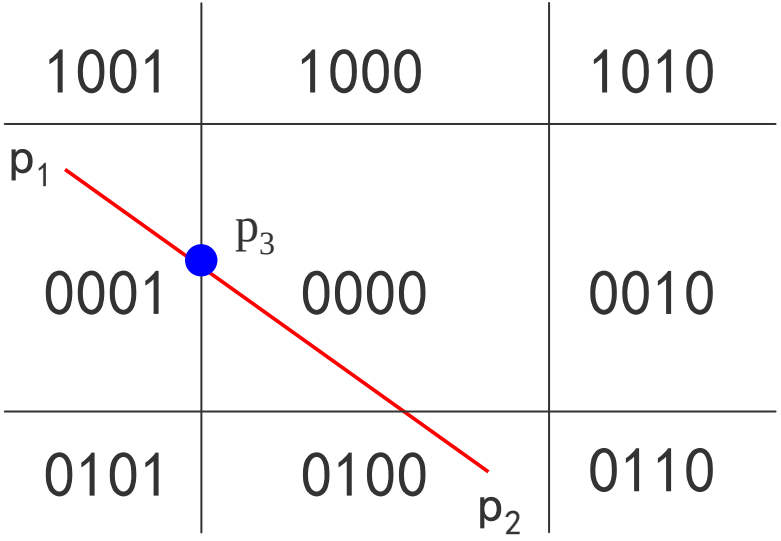
首先对P_1P_2进行编码

或

	0001
	0100
	0101

与

	0001
	0100
	0000

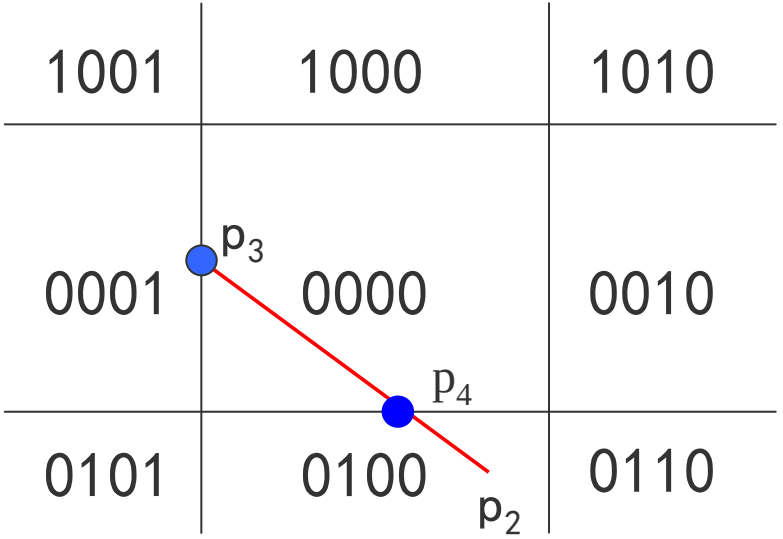


按左、右、下、上的顺序求出直线段与窗口左边界的交点为P_3，P_1P_3必在窗口外，可简弃

对P_2P_3重复上述处理

或
0000
0100
0100

与
0000
0100
0000



剩下的直线段（P_3P_4）再进行进一步判断， $code_1|code_2=0$，全在窗口中，简取之。

小 结

Cohen-Sutherland算法用编码的方法实现了对直线段的裁剪

编码的思想在图形学中甚至在计算机科学里也是非常重要的，一个很简单的思想可以带来很了不起的作用。

比较适合两种情况：一是大部分线段完全可见；二是大部分线段完全不可见。