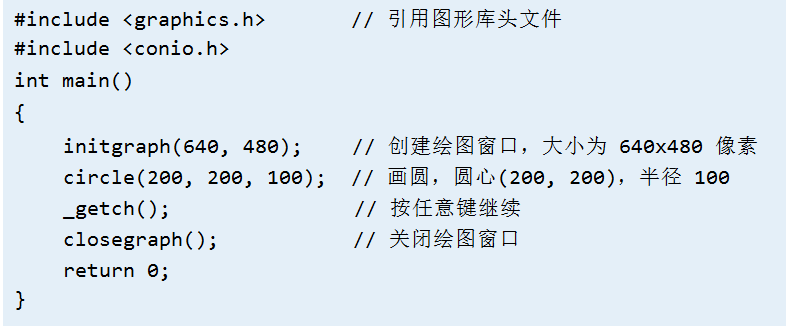
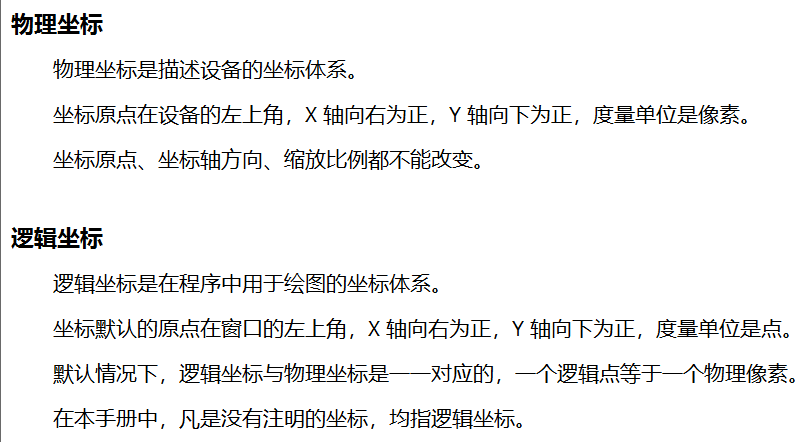
#基本操作：



#坐标：



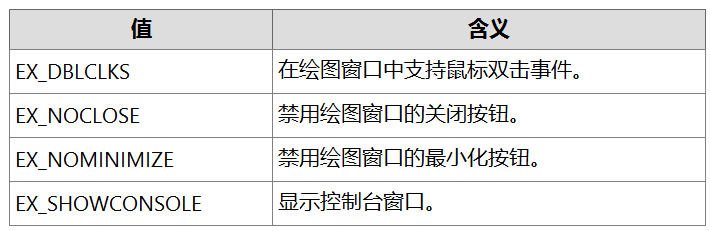
#绘图设备相关函数：

1.void cleardevice();使用当前背景色清空设备；

2.void closegraph();关闭绘图窗口；

3.HWND initgraph( int width, int height, int flag = NULL

);用于初始化绘图窗口，参数为宽度、长度、绘图窗口样式，flag可以为：





4.void getaspectratio( float \*pxasp, float \*pyasp

);获取缩放因子，参数分别是x和y方向的缩放因子；

5.void setaspectratio( float xasp, float yasp

);设置缩放因子，参数分别是x和y方向的缩放因子；

注：如果缩放因子为负数，则可以实现坐标轴的反转。

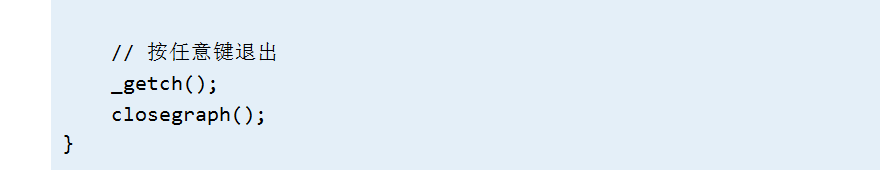
6.void setorigin(int x, int y);设置坐标原点，使用物理坐标；

7.void setcliprgn(HRGN hrgn);设置裁剪区，使用物理坐标；

void clearcliprgn();用于清空裁剪区；







#图形颜色及其样式：

注：样式部分没有写，比较抽象；

1.COLORREF getbkcolor();获取当前设备背景色；

2.COLORREF getfillcolor();获取当前设备填充颜色；

3.COLORREF getlinecolor();获取当前设备画线颜色；

4.COLORREF getpixel(int x,int y);获取点的颜色；

5.void setbkcolor(COLORREF color);设置当前设备绘图背景色；

6.void setfillcolor(COLORREF color);设置当前设备填充颜色；

7.void setlinecolor(COLORREF color);设置当前设备画线颜色；

#图形绘制相关函数：

注：在12345中：

前缀fill:画填充图形；

前缀solid:无边框的填充图形；

前缀clear:清空图形区域；

1.void circle( int x, int y, int radius );画圆；

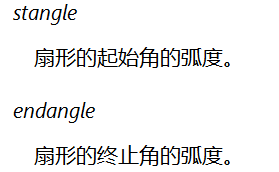
2.void ellipse( int left, int top, int right, int bottom);画椭圆；

特有：void arc( int left, int top, int right, int bottom, double stangle, double endangle );画椭圆弧，前四个参数为左上角和右下角，第五个为圆弧起始角的弧度，第六个为圆弧终止角的弧度；

3.void polygon( const POINT \*points, int num);画多边形；points是每个点的坐标，数组元素个数为num；

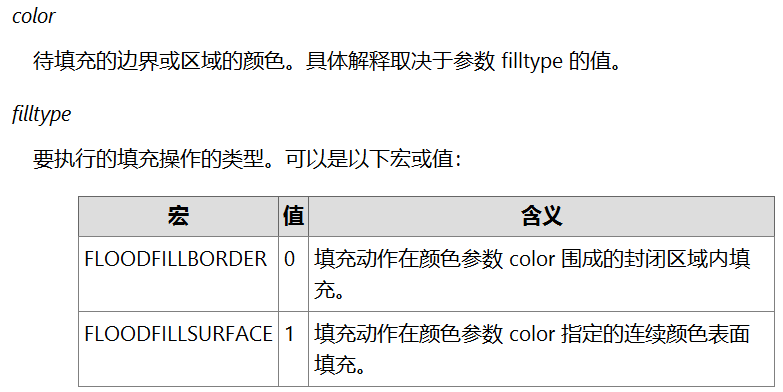
4.void rectangle( int left, int top, int right, int bottom);画矩形；

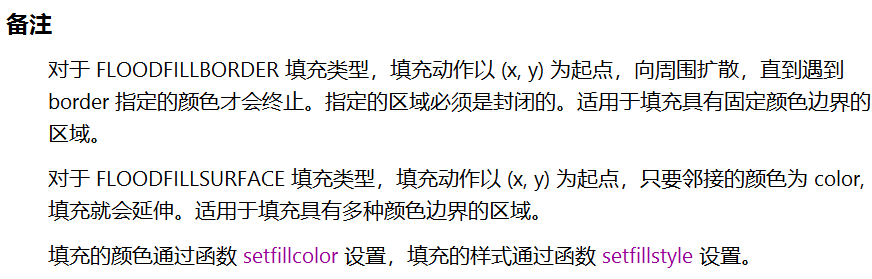
5.void pie( int left, int top, int right, int bottom, double stangle, double endangle);画扇形；



6.void floodfill( int x, int y, COLORREF color, int filltype = FLOODFILLBORDER

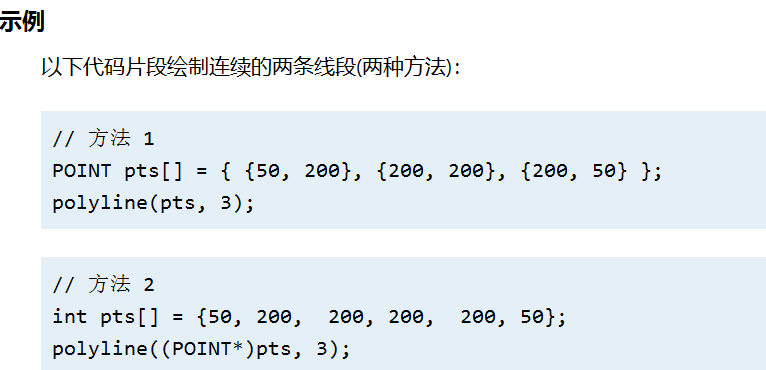
);填充区域，从(x,y)开始延伸直至边界；





7.void line( int x1, int y1, int x2, int y2);画直线（线段）；

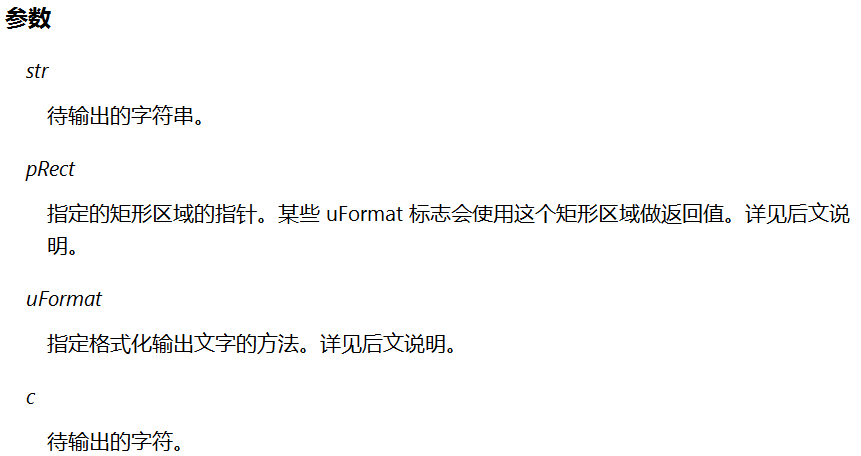
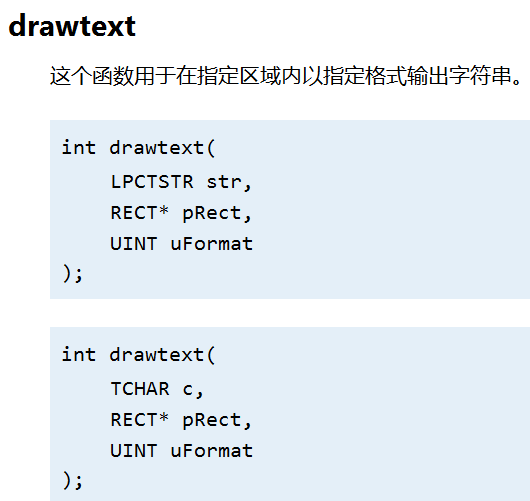
8.void polyline( const POINT \*points, int num);画折线；

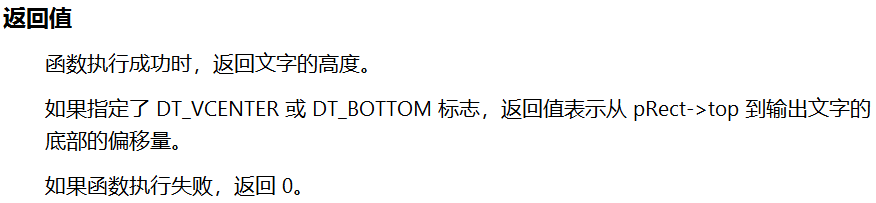


9.void putpixel( int x, int y, COLORREF color);画一个点；

#文字输出相关函数：

1.





uFormat的常用格式：

DT\_CENTER水平居中；DT\_LEFT左对齐；DT\_RIGHT右对齐;DT\_TOP顶部对齐；

DT\_WORDBREAK自动换行；

DT\_EXPANDTABS展开Tab符号；

DT\_NOCLIP使输出文字不受pRect裁剪限制，会使这个函数执行速度变得稍快一点；

DT\_RTLREADING设置从右到左阅读方式；

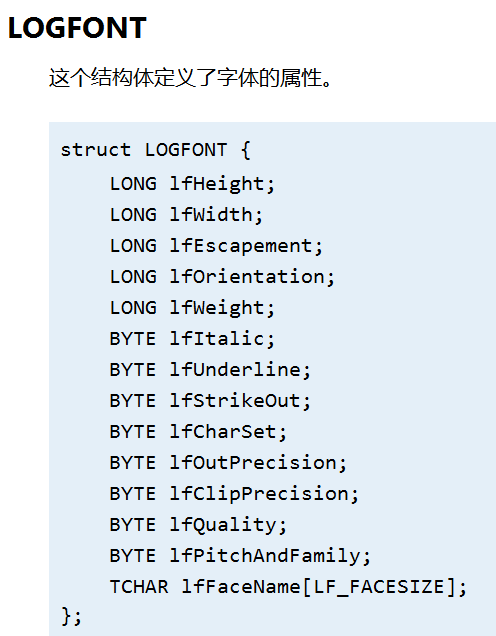
DT\_VCENTER|DT\_SINGLELINE文字垂直居中；

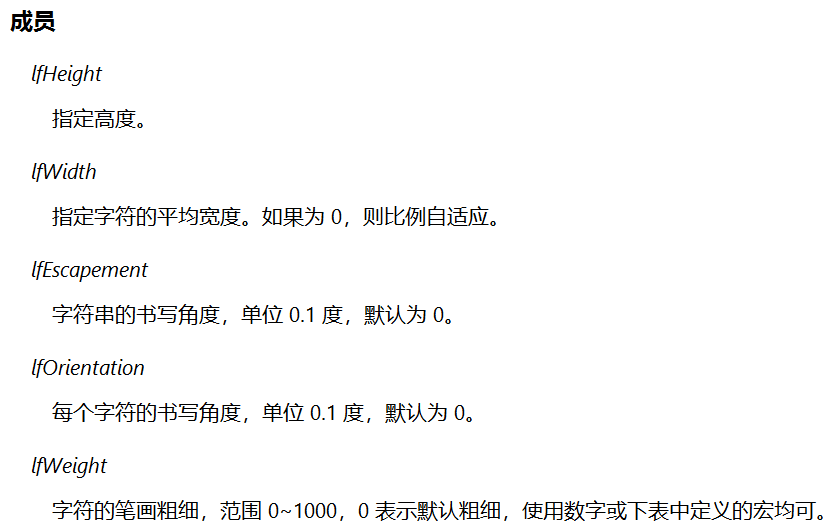


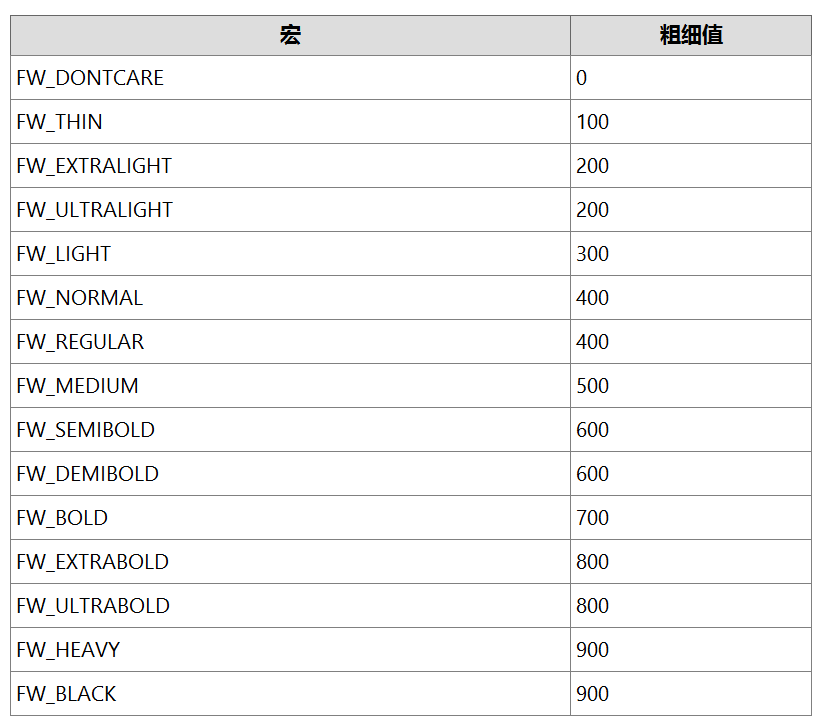
2.COLORREF gettextcolor();获取文字颜色；

3.void gettextstyle(LOGFONT \*font);获取文字样式，font是指向LOGFONT结构体的指针；

4.



、





其他的没什么用了；

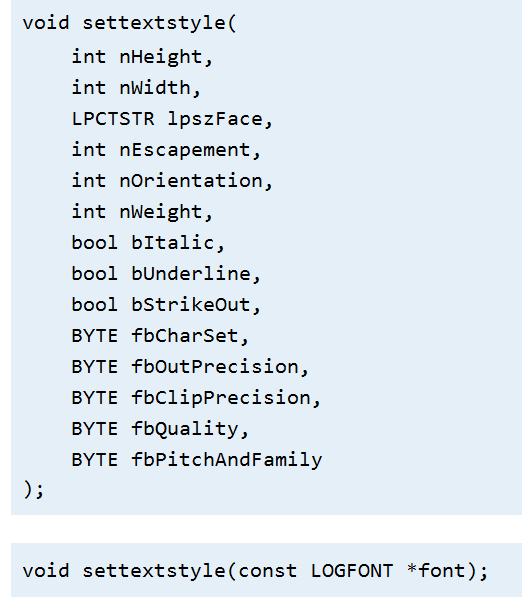
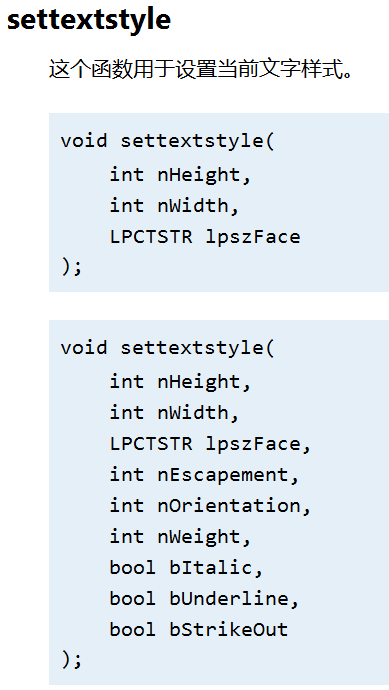
5.void outtextxy( int x, int y, LPCTSTR str);

void outtextxy( int x, int y, TCHAR c);

在指定位置输出字符串或者字符；

6.void settextcolor(COLORREF color);设置当前文字颜色；

7.

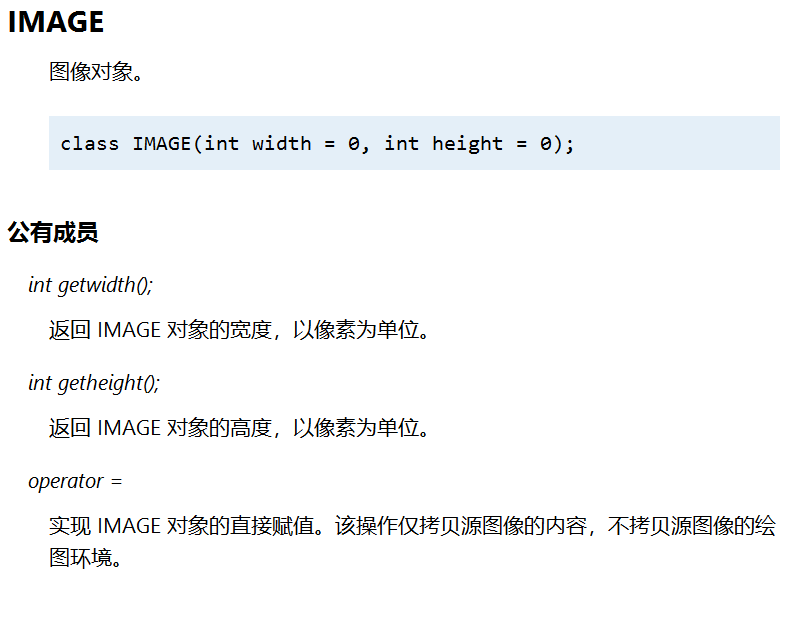


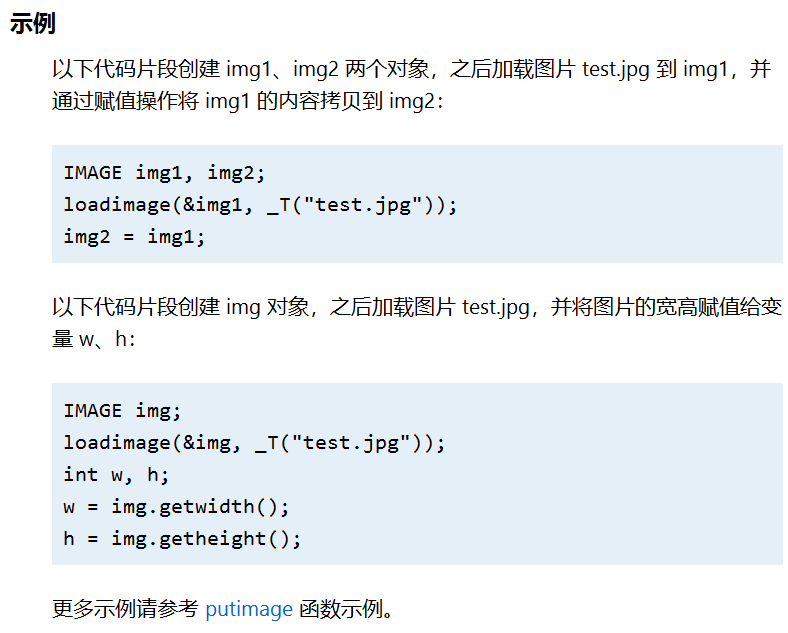




#图像处理相关函数：

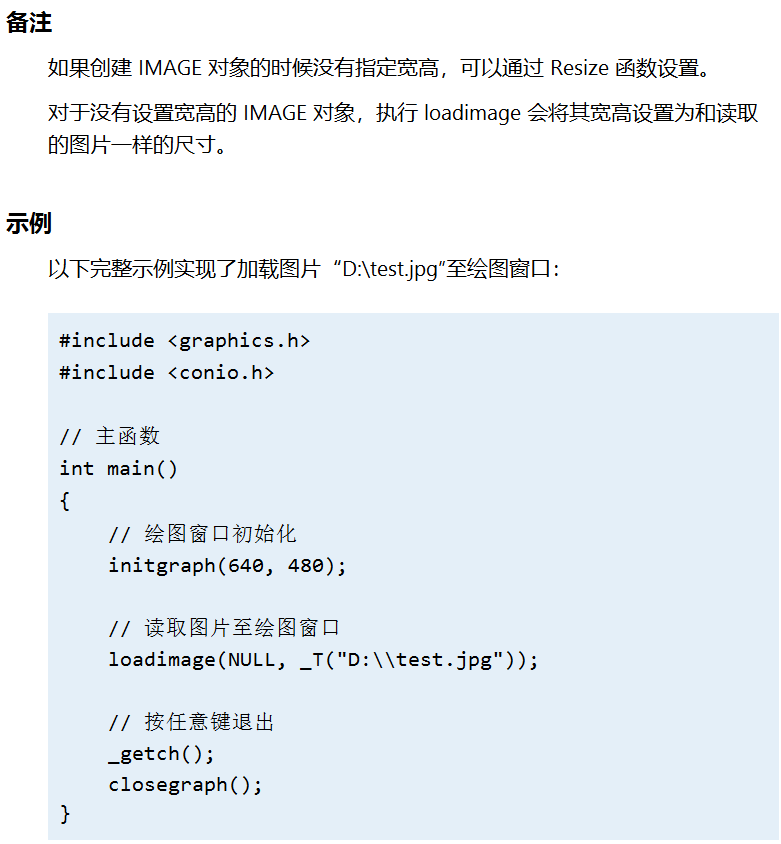
1.





2.

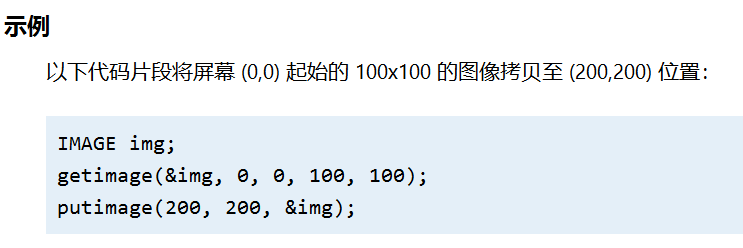






3.





4.

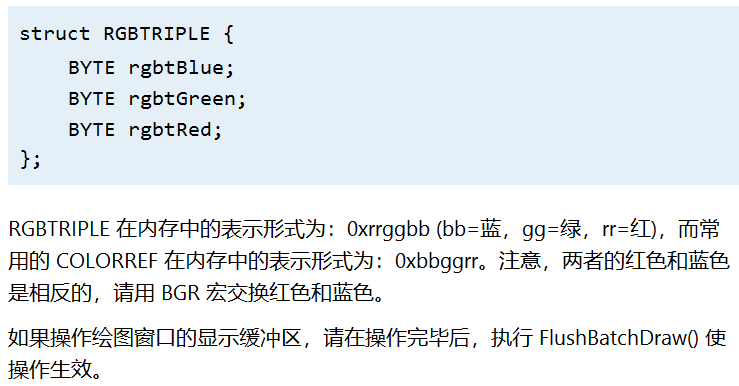


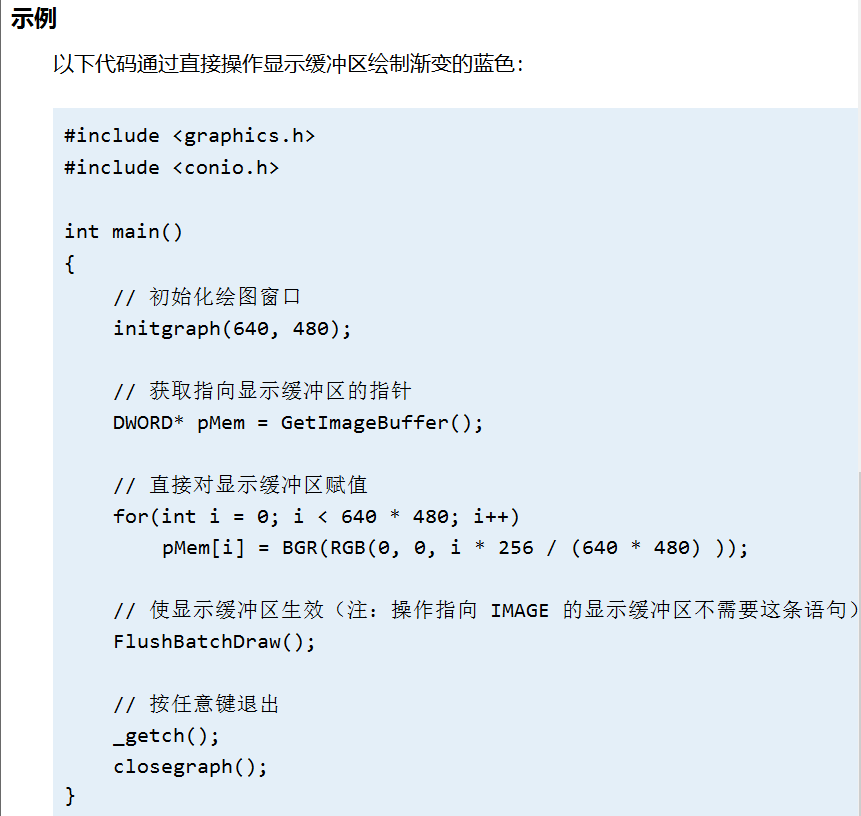
5.



6.

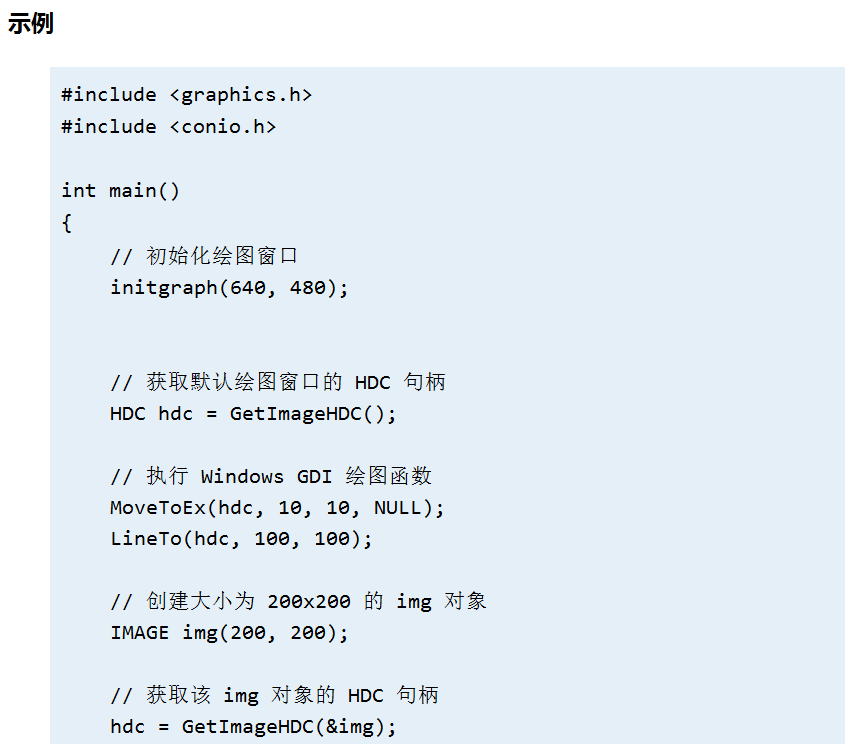


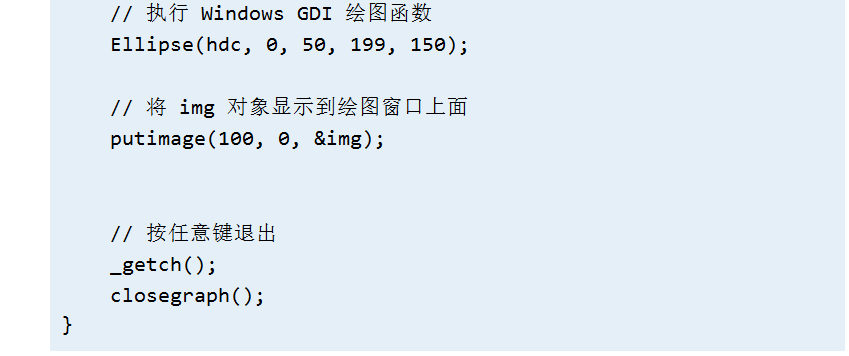




7.

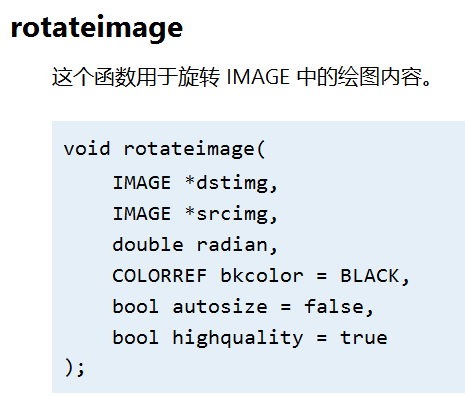


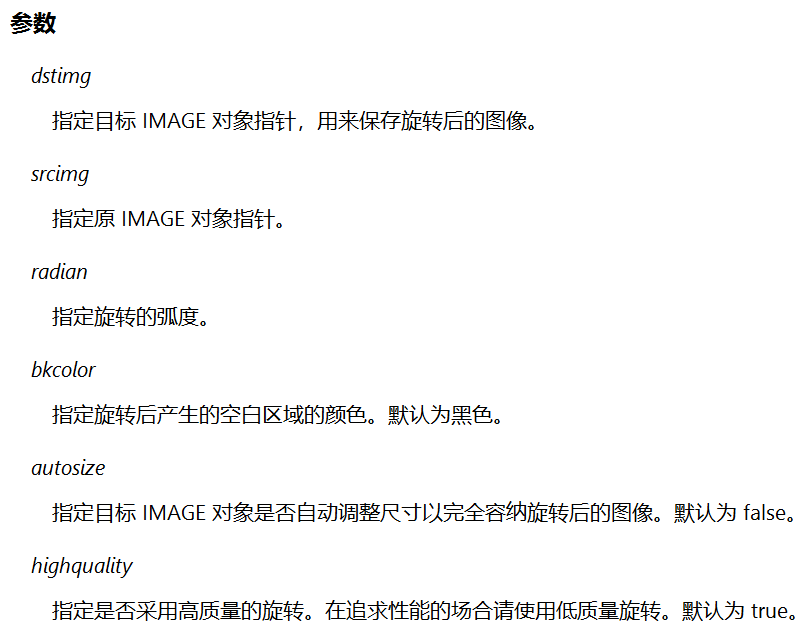


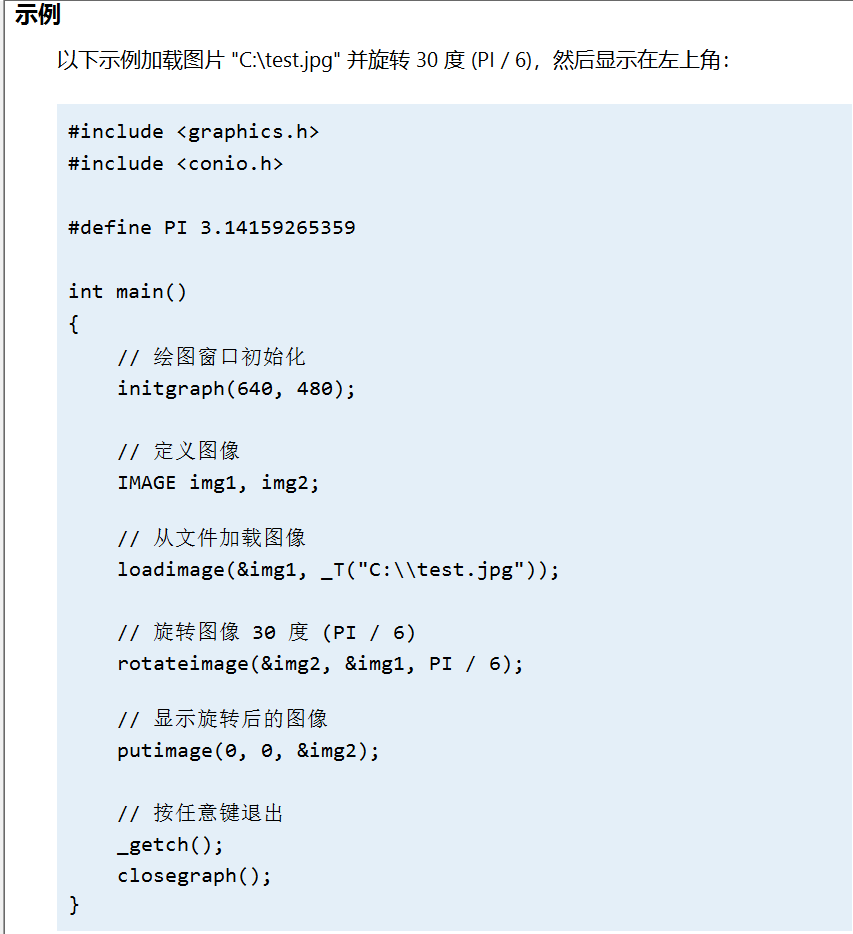


8.IMAGE\* GetWorkingImage();获取当前绘图设备；

9.







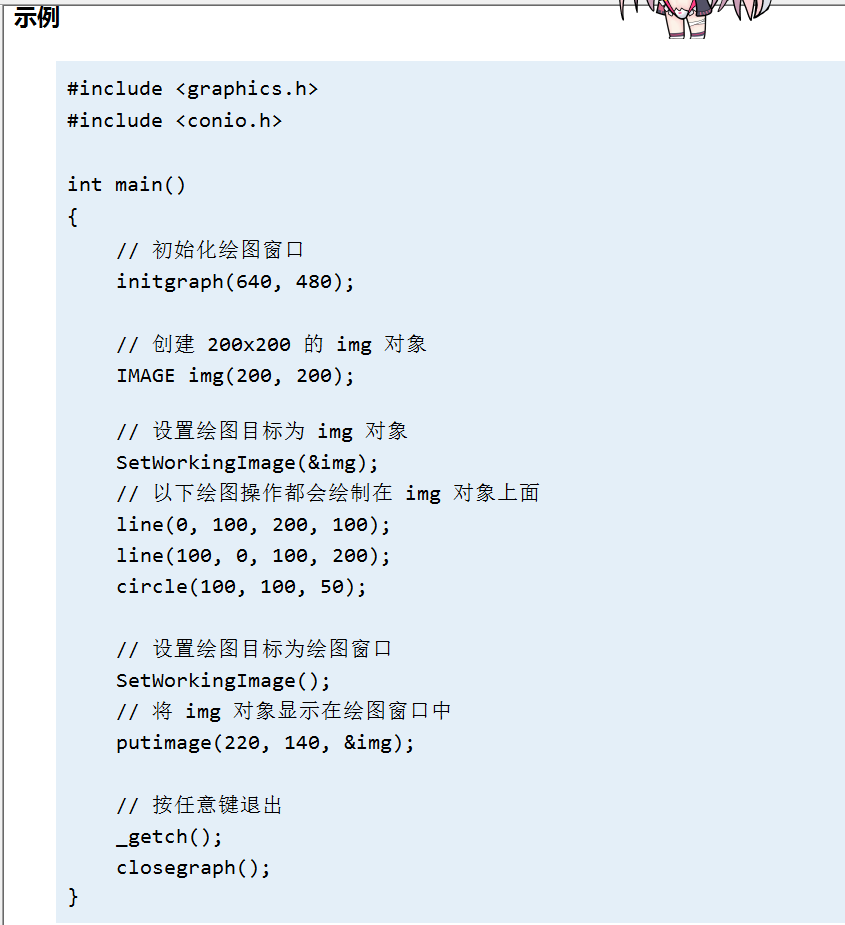
10.





11.

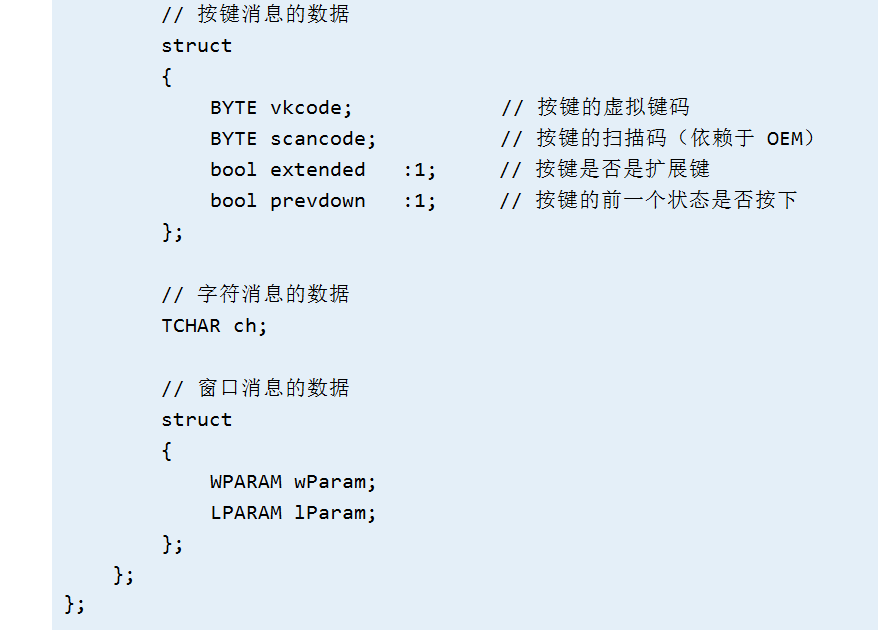


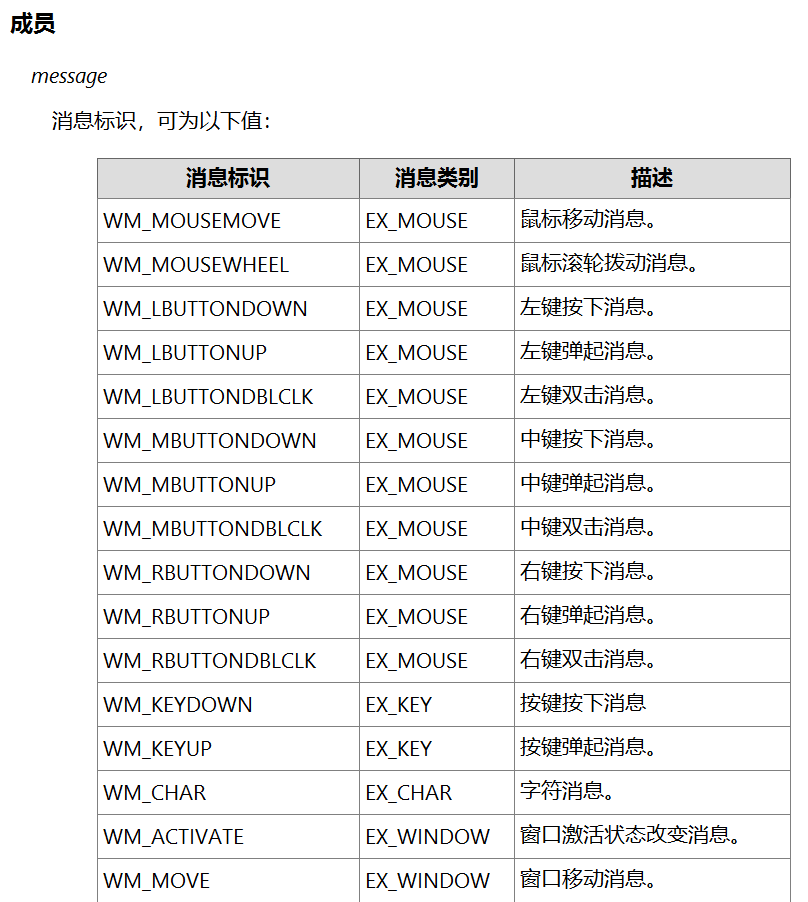


#信息处理相关函数：

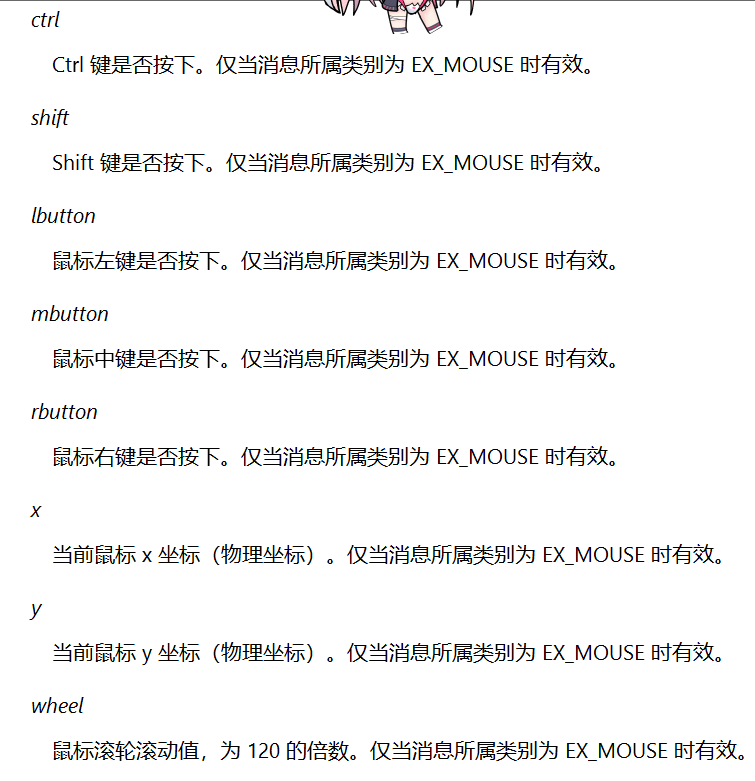
1.











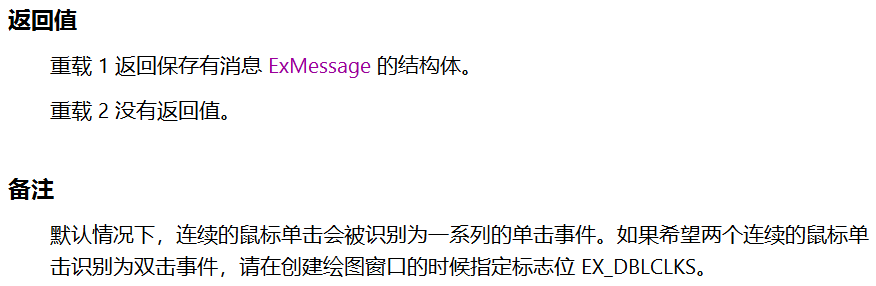


2.



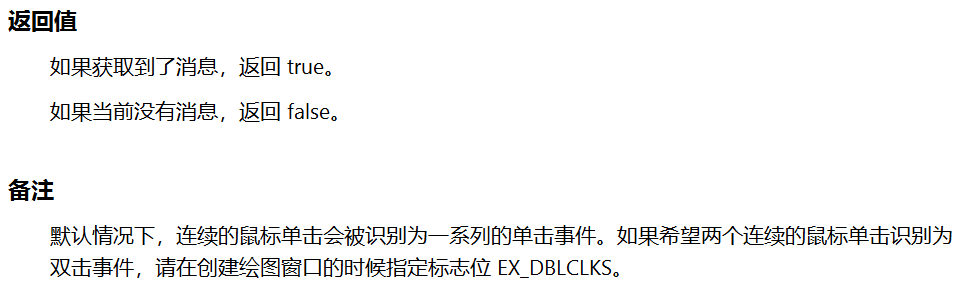
3.





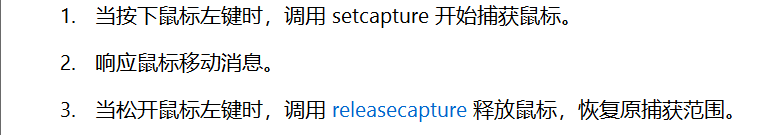
4.



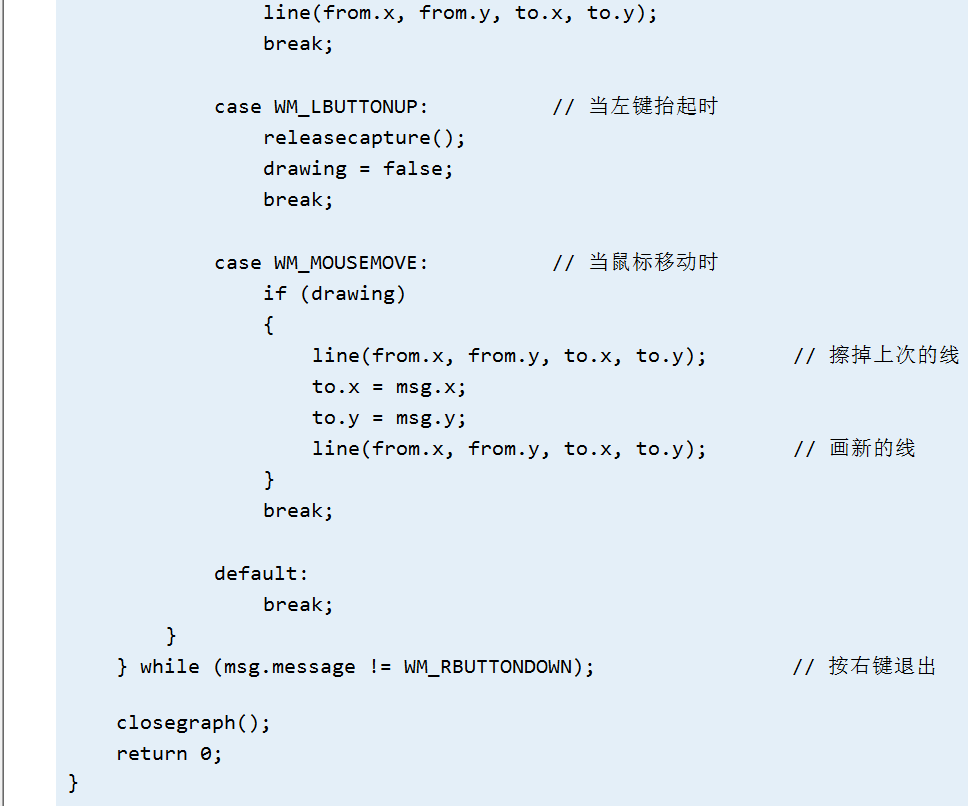


5.

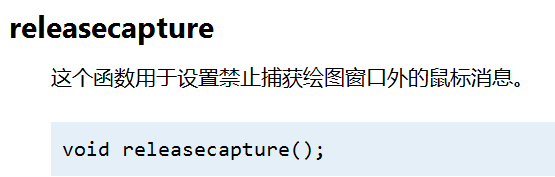








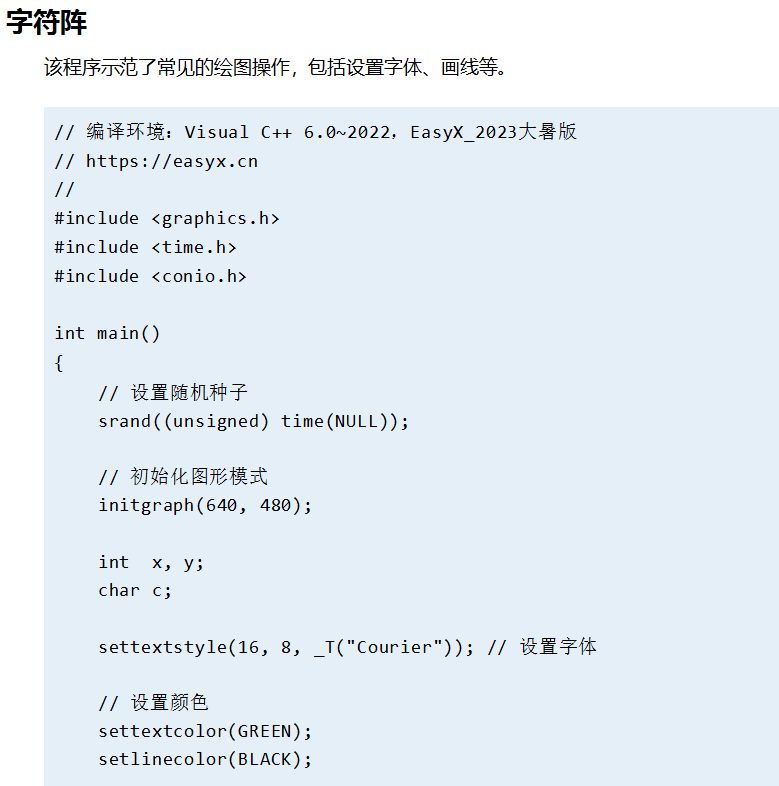
6.

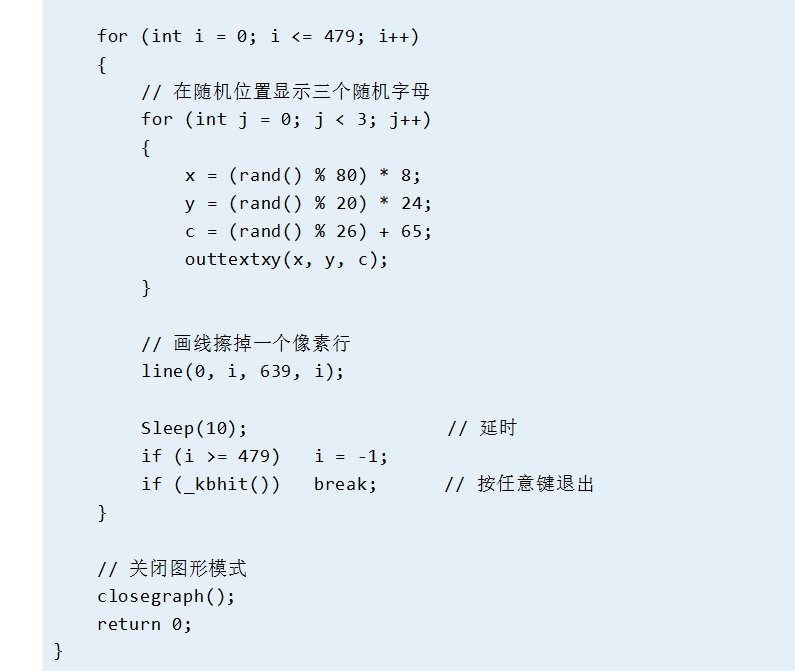


#其他函数：



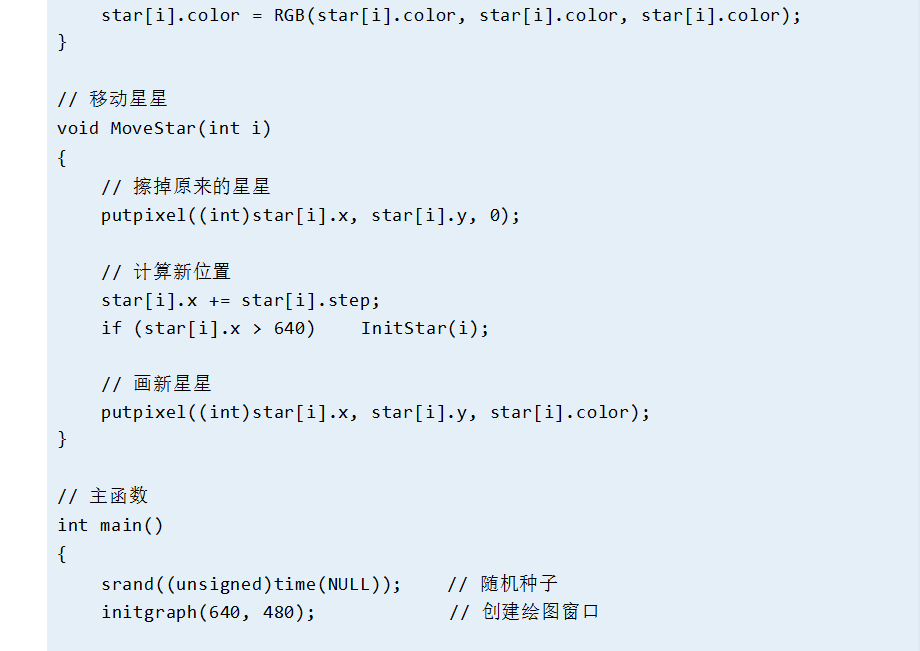
#示例程序1：字符阵：

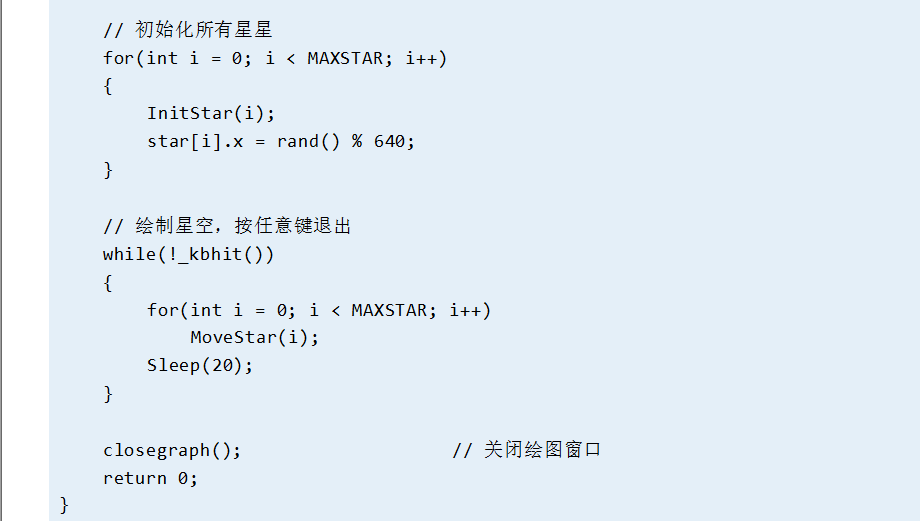




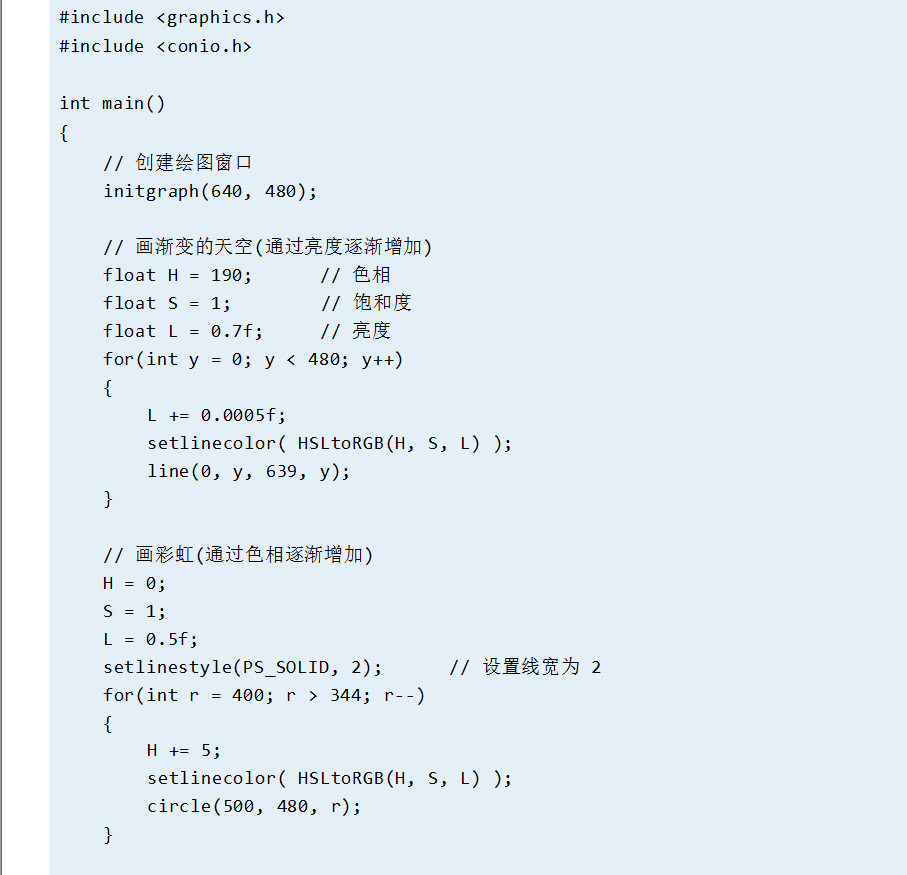
#示例程序2：星空：

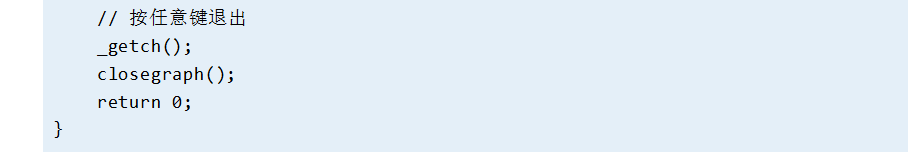






#示例程序3：彩虹：





#示例程序4：鼠标操作：



