
EGE图形库帮助文档 -
Made by UNREGISTERED
version of Easy CHM

Table of Contents

- 1. 主页 6
- 2. 介绍 7
 - 2.1 EGE基本说明 7
- 3. 安装 9
 - 3.1 EGE安装 9
- 4. 使用预览 15
 - 4.1 使用预览 15
- 5. 函数 17
 - 5.1 库函数目录 17
 - 5.2 颜色表示及相关函数 18
 - 5.2.1 颜色表示及相关函数 18
 - 5.2.2 颜色表示 19
 - 5.2.3 EGEGET B..... 19
 - 5.2.4 EGEGET G 20
 - 5.2.5 EGEGET R..... 20
 - 5.2.6 EGERGB..... 21
 - 5.2.7 getbkcolor 22
 - 5.2.8 getcolor 22
 - 5.2.9 getfillcolor 23
 - 5.2.10 hsl2rgb 23
 - 5.2.11 hsv2rgb 24
 - 5.2.12 rgb2gray 25
 - 5.2.13 rgb2hsl 26
 - 5.2.14 rgb2hsv 27
 - 5.2.15 setbkcolor..... 28
 - 5.2.16 setbkcolor f 29
 - 5.2.17 setbkmode 29
 - 5.2.18 setcolor 30
 - 5.2.19 setfillcolor 30
 - 5.2.20 setfontbkcolor 31
 - 5.3 绘制图形相关函数 32
 - 5.3.1 绘制图形相关函数 32
 - 5.3.2 arc 33
 - 5.3.3 bar 35
 - 5.3.4 bar3d 36
 - 5.3.5 circle 37
 - 5.3.6 drawbezier 38
 - 5.3.7 drawlines 39
 - 5.3.8 drawpoly..... 40
 - 5.3.9 ellipse..... 41
 - 5.3.10 fillellipse 43

5.3.11 fillpoly	44
5.3.12 floodfill	46
5.3.13 floodfill	47
5.3.14 getfillcolor	48
5.3.15 getheight	49
5.3.16 getlinestyle	49
5.3.17 getpixel	50
5.3.18 getwidth	51
5.3.19 getx	51
5.3.20 gety	52
5.3.21 line	53
5.3.22 linerel.....	54
5.3.23 lineto.....	55
5.3.24 moverel	56
5.3.25 moveto	56
5.3.26 pieslice.....	57
5.3.27 putpixel.....	58
5.3.28 putpixels	59
5.3.29 rectangle	59
5.3.30 sector	60
5.3.31 setfillcolor	61
5.3.32 setfillstyle	62
5.3.33 setlinestyle	63
5.3.34 setlinewidth	64
5.3.35 setwritemode.....	65
5.4 绘图环境相关函数	66
5.4.1 绘图环境相关函数	66
5.4.2 cleardevice	66
5.4.3 clearviewport	67
5.4.4 closegraph	68
5.4.5 gettarget	68
5.4.6 getviewport	69
5.4.7 initgraph	70
5.4.8 is run	71
5.4.9 setactivepage	71
5.4.10 setcaption.....	72
5.4.11 setinitmode	72
5.4.12 setrendermode	73
5.4.13 settarget	73
5.4.14 setviewport	74
5.4.15 setvisualpage	75
5.4.16 window getviewport	76
5.4.17 window setviewport.....	77
5.5 文字输出相关函数	78

5.5.1 文字输出相关函数	78
5.5.2 getfont	78
5.5.3 logfont	79
5.5.4 outtext	83
5.5.5 outtextrect	84
5.5.6 outtextxy	85
5.5.7 setfont	86
5.5.8 setttextjustify	89
5.5.9 textheight	89
5.5.10 textwidth	90
5.5.11 xyprintf	90
5.5.12 xyprintf	91
5.6 图像处理相关函数	92
5.6.1 图像处理相关函数	92
5.6.2 getimage	93
5.6.3 imagefilter blurring	94
5.6.4 PIMAGE.....	95
5.6.5 putimage	96
5.6.6 putimage alphablend	99
5.6.7 putimage alphantransparent	100
5.6.8 putimage transparent.....	101
5.6.9 三元光栅操作码	103
5.7 键盘鼠标输入函数	112
5.7.1 键盘鼠标输入函数	112
5.7.2 FlushMouseMsgBuffer	112
5.7.3 getch	113
5.7.4 getkey	113
5.7.5 getmouse.....	114
5.7.6 GetMouseMsg	115
5.7.7 kbhit	115
5.7.8 kbmsg	116
5.7.9 key_msg 结构体	117
5.7.10 keystate	118
5.7.11 mouse_msg 结构体	118
5.7.12 mousemsg	119
5.7.13 MOUSEMSG 结构体	120
5.7.14 mousepos	121
5.7.15 showmouse	122
5.8 数学函数	122
5.8.1 数学函数	122
5.9 其它函数	123
5.9.1 其它函数	123
5.9.2 getfps	123
5.9.3 GetHWnd	124

5.9.4 inputbox getline	125
5.10 随机函数	126
5.10.1 随机函数	126
5.10.2 random	126
5.10.3 randomf	127
5.10.4 randomize	127
5.11 时间函数	128
5.11.1 时间函数	128
5.11.2 api sleep	128
5.11.3 delay	129
5.11.4 delay fps	129
5.11.5 delay jfps	130
5.11.6 delay ms	130
5.11.7 fclock	131
6. 入门基础示例	131
6.1 入门基础示例	131
6.2 第一个画图程序	133
6.3 第一个图形窗口程序	133
6.4 高级：绘画渐变色 / 彩虹	134
6.5 更多的文字对齐方式设置	134
6.6 更多的颜色表示办法 (HSV/HSL)	134
6.7 基础动画八，使用 C++封装你的对象	135
6.8 基础动画二：简单平移动画	135
6.9 基础动画九，使用场景，更灵活地控制动画	135
6.10 基础动画六，利用对象封装，显示多个对象	136
6.11 基础动画七，使用非整数参数，随机色彩，使你的动画更自然	136
6.12 基础动画三，更丰富的变化与帧率	136
6.13 基础动画十，场景管理下的输入输出控制	137
6.14 基础动画四，半透明淡入淡出	137
6.15 基础动画五，对象封装，使代码更有条理	137
6.16 基础动画一：规范框架	138
6.17 屏幕背景色	138
6.18 色彩的计算	138
6.19 使用 IMAGE——透明，半透明	138
6.20 使用 IMAGE——图片缩放	139
6.21 使用 IMAGE——抓图，绘画	139
6.22 图形边界和填充	139
6.23 图形视口裁剪	139
6.24 文字绘制与文字相关设置	140
6.25 颜色填充	140
6.26 用户交互——键盘	140
6.27 用户交互——鼠标 1	141
6.28 用户交互——鼠标 1	141
6.29 用户交互——鼠标 2	141

- 6.30 用户交互——字符串数据输入 141
- 7. 示例程序 142
 - 7.1 ege 示例程序 142
- 8. 网友程序 143
 - 8.1 ege 网友小程序 143
- 9. 关于本 EGE图形库 144
 - 9.1 关于本 EGE图形库 144

1. 主页

EGE(Easy Graphics Engi13.03

作者： misakamm

官网：[tp://xege.org](http://xege.org)

主页 1：<http://miamm.github.com/xege>

主页 2：[h://misakamm.bitbucket.org/index.htm](http://misakamm.bitbucket.org/index.htm)

主页 3：<http://tcgraphics.sourceforge.net>

主页 4：

<http://tcgraphics.svn.sourceforge.net/viewvc/tcgraphics/trunk/index.htm>

项目主页 <https://sourceforge.net/projects/tcgraphics>

博客：<http://blog.misakamm.org>

邮件列表订阅：<https://lists.sourceforge.net/lists/listinfo/tcphics-faqs>

库下载：

[ege13.03 bitbucket](#) [下载](#)

[ege13.03 gitcafe](#) [下载](#)

[ege13.03 sourceforge](#) [下载](#)

方法 2：请向 [moegirl at misakamm dot org](mailto:moegirl@misakamm.org) 发送主题为 ege 的邮件，会自动给你回复 32 位版本下载链接

库内已包含离线文

关于 32 位和 64 位：注意不是你的操作系统是多少位就下载多少位，而是你的编译器是多少位就用多少位，或者能支持多少位就用多少位。

如果是 MinG 那么请用 32 位的，如果是 vc9/vc10，那可以两个都下载。如果你不知道，两个都下载就没错了，
不过，目前多数情况下都是 32 位。

项目相关下 <https://sourceforge.net/projects/tcaphics/files/>

库教程和文档目录（注意：本文档对应的 ege 版本为 13.03）：

[基本说明 \(See 2.1\)](#)

[图形库安装说明 \(See 3.1\)](#)

[使用预览 \(See 4.1\)](#)

[入门基础示例和教程](#) (See 6.1)

[库函数目录](#) (See 5.1)

[示序](#) (See 7.1)

[网友小程序（新！有兴趣可投稿哦！）](#) (See 8.1)

[关于本图形库](#) (See 9.1)

如果以上目录文字显示较小， 请用 Ctrl+ 鼠标滚轮调整文字大小，或者用你的浏览器设置修改即可

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

2. 介绍

2.1 EGE 基本说明

[主页](#) (See 1.)- > EGE基本说明

本库是一个类似 `graphics.h` 的 BGI 图形库，使用方法相当接近，所习过 TC 图形的话会很容易上手

基本使用方法几乎和 TC 的一样，只要 `#include "graphics.h"` 就可以了

本图形库目前支持 VC6, C2008, VC2010 等 IDE，同时也支持 MinGW 编译器和使用 MinGW 编译器的 IDE（如 Code::Blocks，C-Free，Dev-C，eclipse for C/C++）
在 `graphics.h` 头文件的最顶处有相关的说明，并且附带较详细的 pdf 文档

为什么要写这个库？

许多学编程的都是从 C 语言开始入门的，而目前的现状是：

1. 有些学校以 Turbo C 为环境讲 C 语言，只是 Turbo C 的环境实在太老了，复制粘贴都很不方便。
2. 有些学校直接拿 VC 来讲 C 语言，因为 VC 的编辑和调试环境都很优秀，并且 VC 有适合教学的免费版本。可惜在 VC 下只能做一些文字性的练习题，想画

条直线都很难，还要注册窗口类、建消息循环等等，初学者会受严重打击的。初学编程想要绘图就得用 TC，很无奈。

3. 还有计算机图形学，这门课程的重点是绘图而不是 Windows 编程。所以，许多老师不得不用 TC 教学，因为 Windows 绘图太复杂了，会偏离教学的重点。新的图形学的书有不少是用的 OpenGL，可是门槛依然很高。

所以，我想给大家一个更好的学习平台，就是 VC 方便的开发平台和 TC 简单的绘图功能，于是就有了这个库。刚开始学 C 语言，或者您是一位教 C 语言的老师，又或者您在教计算机图形学，那么这个库一定会让您兴奋的。

另外：这个库不是为了给你直接移植 BGI 码用的，直接移植多半是不能通过编译的。

如果真需要移植，请做相应的修改，或者使用高兼容性的 ege 老版本图形库

本图形库的优点：

绘图效率较好，特别在批量绘图模式下，640*480 的半透明混合，可以直接使用 getpixel / putpixel 完成，

并且可以在大约 1.5G CPU 台式机器上达到 60fps（要否则一般的家用机再强也不行）

灵活性更强，绘图可以直接针对一个 PIMAGE，不必只能画在屏幕上

功能更多，支持拉伸贴图，支持透明半透明贴图，图像模糊滤镜操作，

可以用对话框函数进行图形化的输入

面向动画编程和游戏编程做了功能增强，可以方便地对帧率进行准确的控制

附带简单的 3 运算函数和类，便于 3D 计算

支持读写多种格式图片，支持读取 bmp, jpg, png, gif，支持保存为 bmp 和 png

支持 GUI 模式编

简要安装说明

把 include/graphics 文件，复制到你的编译器安装目录下的 include 目录内
再把 lib 目录下的文件，复制到你的编译器安装目录下的 lib 目录内，具体编译器所依赖的文件情况：

VC6: graphics.lib

VS2008: graphics08.lib graphics08d.lib

VS2010: graphics08.lib graphics10d.lib

MinGW: libgraphics.a

简要使用说明

目前模拟了绝大多数 BGI 的绘图函数。使用上，基本和 TC / C 没太大区别。看一个画圆的例子吧：

```
#include "graphics.h" // 引用 ege 图形库
```

```

int main()
{
    nitgraph (640, 480);    // 初始化一个窗口，这里和 TC 略有区别
    circle (200, 200, 100); // 画圆，圆心 (200, 200) ，半径 100
    getch ();              // 暂停一下等待用户按键
    closegraph ();         // 关闭图形界面
    return 0;
}

```

呵呵，很简单

详细安装和使用方法，请看本文档的安装部分。
 如问题，请直接在 [“ ege 娘 ” 贴吧](#) 发主题帖子吧。
 官方联系邮箱： misakamm[at]gmail[dot]com

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

3. 安装

3.1 EGE 安装

[主页 \(See 1.\)](#) - > EGE安装

[点这里查看视频教程帖子](#)，以下是文件说明：

简要安装

首先把压缩包里的 include 目录下所有文件，复制到你的编译器安装目录下的 include 目录内，不管是哪个编译器
 比如，vc2008 的 include 路径类似 “ Microsoft Visual Studio 9.0 include ”，
 而 lib 路径类似 “ Microsoft Visual Studio 9.0 lib ”，vc2010 和 2008 一样。
 vc6 的话，是 “ Microsoft Visual Studio Include ” 和 “ Microsoft Visual Lib ”

然后再把 lib 目录下对应编译器名目录下的文件，复制到你的编译器安装的 lib 目录内。比如你的是 vc6，那就把 下的复制过去。

如果你还是不明白复制到哪里，那请看这个表达：

```

copy "incc\*.*" to "Microsoft Visual Studio 9.0\VC\include\"
copy "lib\*.*" to "Mosoft Visual Studio 9.0\VC\lib\"

```

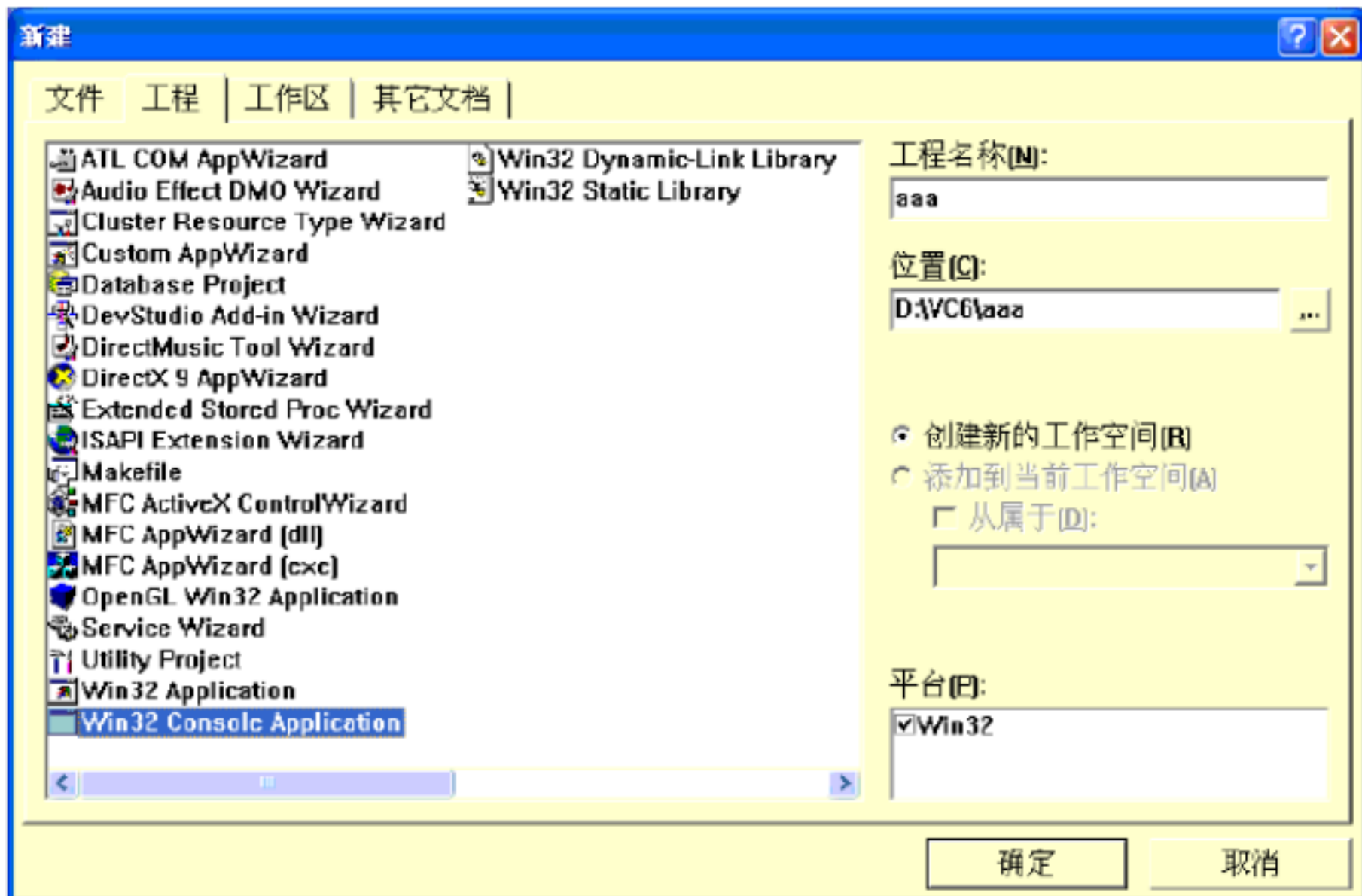
前面的 "include*.*" 是安装包里的文件

复制了这些文件后，就已经成了。

然后，如果是 VC 或者 VS，那么要建立一个工程

以下是 VC6 的操作步骤（但 VS2008/VS2010/VS2012 也类似），请按以下步骤

打开 VC6 后，新建一个 Win32 Console 工程（菜单 -> 文件 -> 新建），如下图：



右上角的工程名字写你喜欢的名字，其下方选择你要建立的工程的目录

点确定后，在弹出的对话框里选择“一个空工程”，然后直接点完成。

然后再次新建，不过这次新建一个 C++ Source File，见下图：



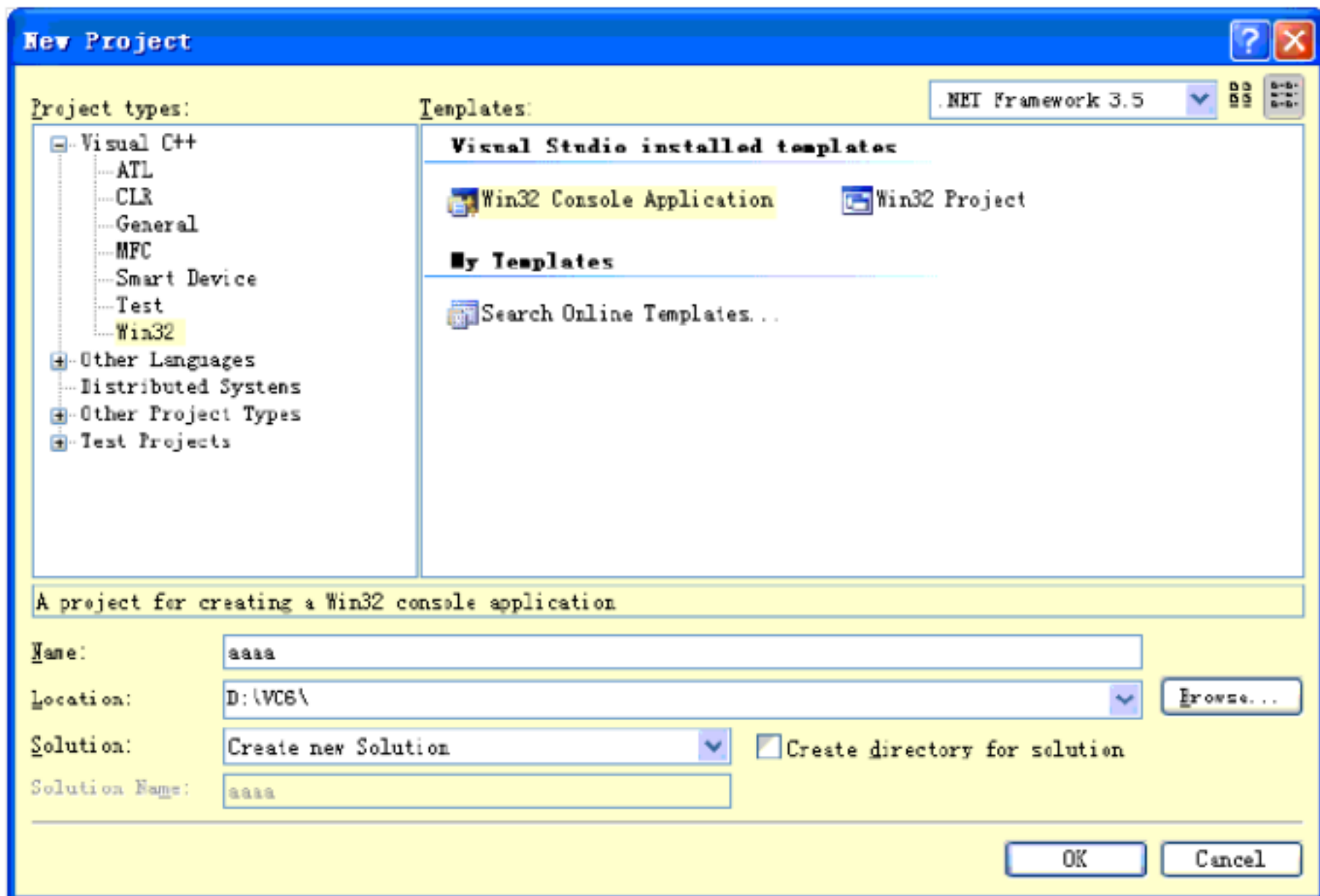
右边文件名写你喜欢建立的名字，然后点确定，就可以开始编写代码了。

最后，请编写如下测试代码检测你的设置是否正确：

```
#include <graphics.h>
int main()
{
    initgraph (640, 480);
    setcolor (GREEN);
    line (100, 100, 500, 200);
    getch ();
    closegraph ();
    return 0;
}
```

然后，按 F7 键，或者点菜单上的 Build （组建）来编译并链接，如果没有错误，就可以按 F5 运行了。

好了，为什么说 VS2008和 VS2010也类似呢？看看如下 VS2008的截图你就明白了：



后面建立文件的时候，在左侧的 General 就能找 +源文件了。如果你的的是中文版，那就细心找找吧，不难找的。

至于 VS2010 的就不截图了，因为几乎和 VS2008 一样。

至于在 Mi 下的编译问题，会麻烦一些，如果你懂使用命令行编译，那么需要用如下指令编译：

```
g++ yourfile.cpp -lgraphics -lgdi32 limm32 -lmsimg32 -lole32 -loleaut32
-lwinmm -luuid -mwindows
```

其中 yourfile.c 你要编译的 cpp 文件

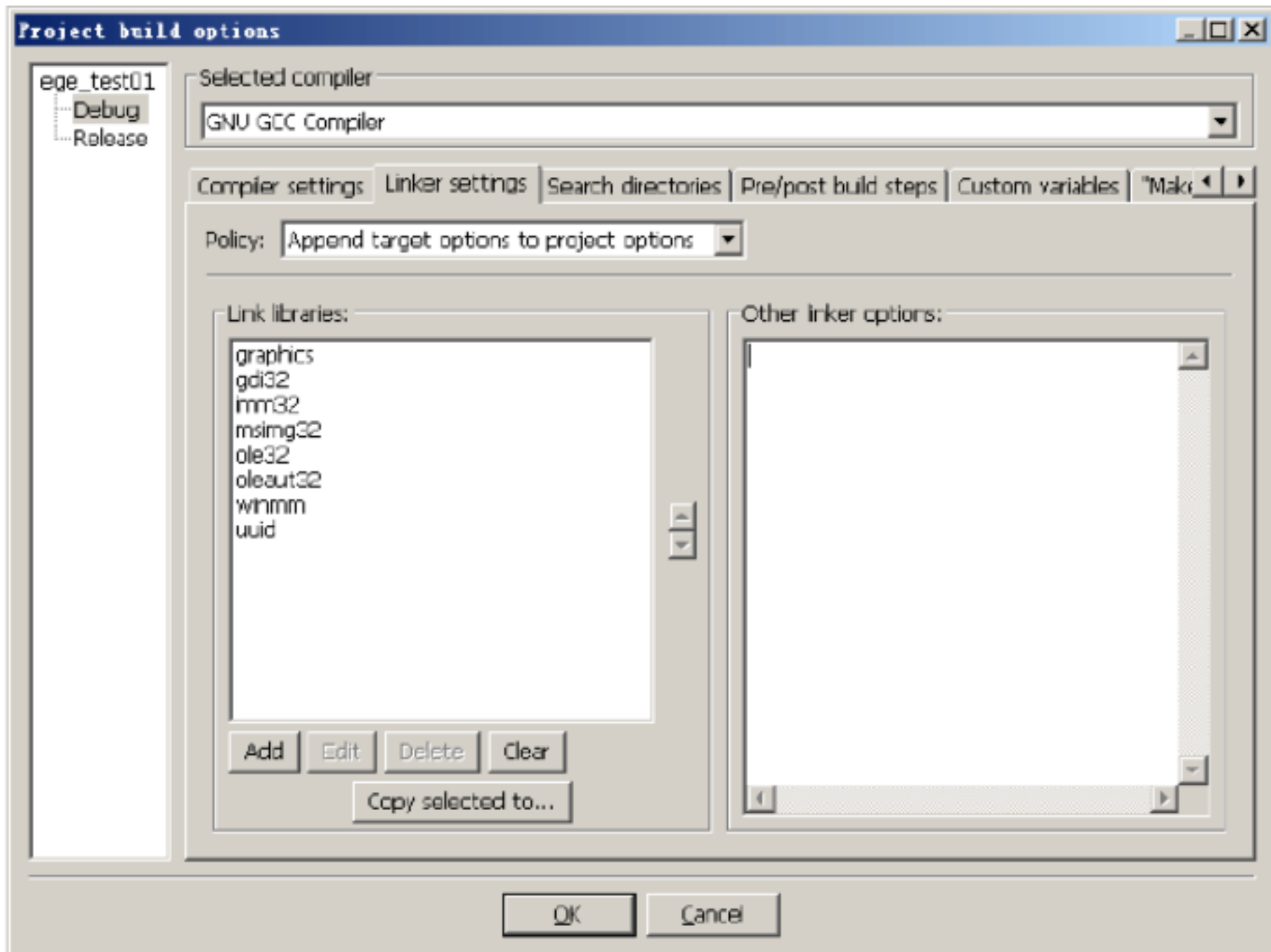
如果你使用的是 IDE，那就要看那个 IDE 是如何设置的，这里无法一一列举，主要需要设置两样东西：

一是建立 Win32 Application 工程（目的是让它以 -mwindows 来编译，这个选项很重要）

二是要添加链接库

需要加入 graphics, gdi32, imm32, msimg32, ole32, oleaut32, winmm, uuid 共 8 个

例如在 Code::Blocks 下（需要建立 project），在菜单 - Project - bui options，设置 Linker 选项，如图：



例如在 C-Free5.0 下，则可 菜单 -> 构建 -> 构建选项 里，设置连接选项，如图：



但是，如果你是以建立工程的方式建立，而不是建立单文件的话，或者你发现上设置无效，那你需要在 菜单 - 工程 - 设置，在这设置连接

此外，C-Free 比较特殊的一点是，如果需要编译为 Win32 应用程序，那么要用 WinMain 来声明你的主函数，如：

```
#include <graphics.h>
int WinMain() // 这里在 graphics.h 里已经定义了宏自动把它展开成合法的声明，以减免声明的麻烦，同时用来欺骗 C-Free
{
    initgraph (640, 480);
    getch ();
    closegraph ;
    return 0;
}
```

又或者，如果你不喜欢这样，那就在链接选项里（即刚刚截图参数框里）手工加上 -mwindows 也可

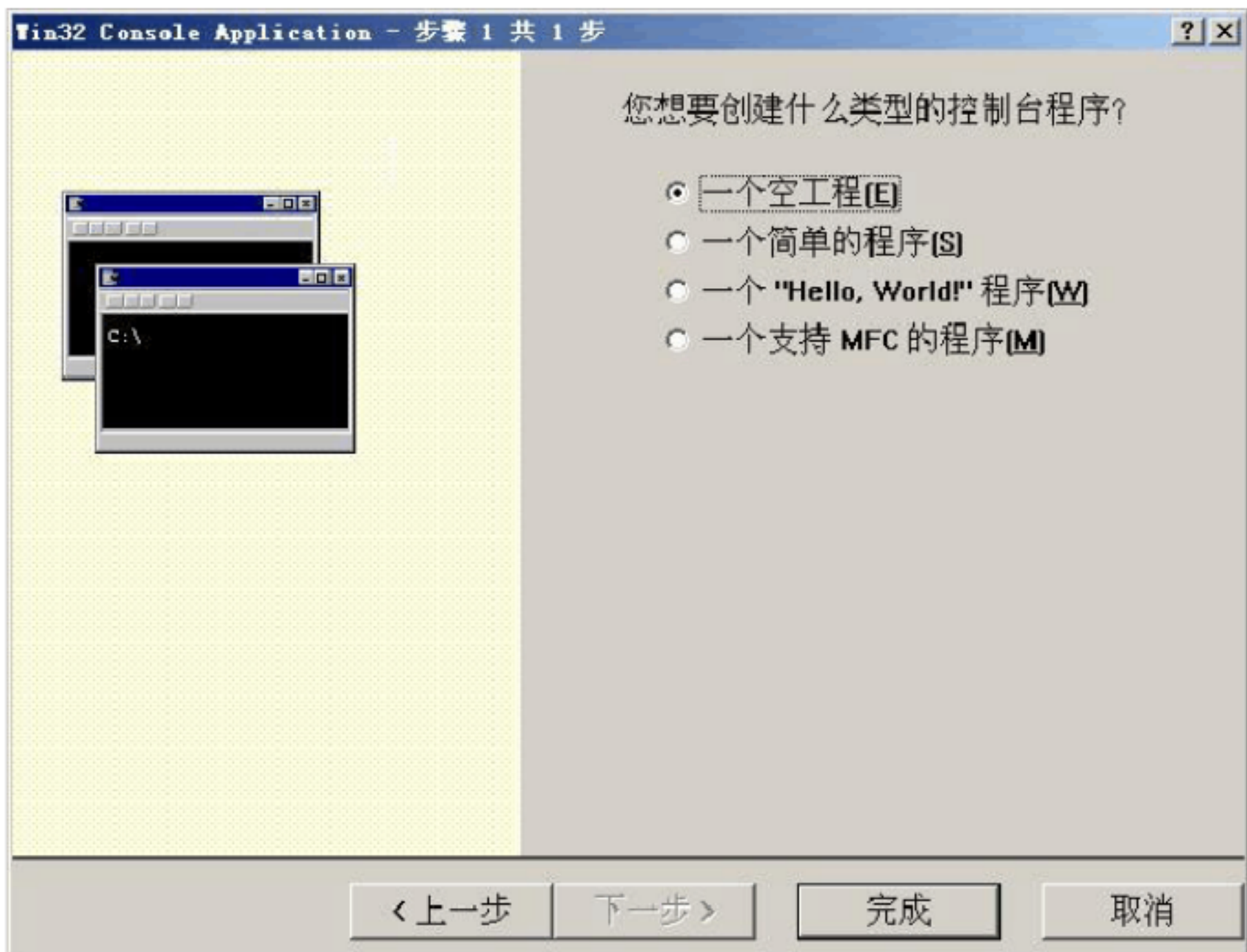
Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

4. 使用预览

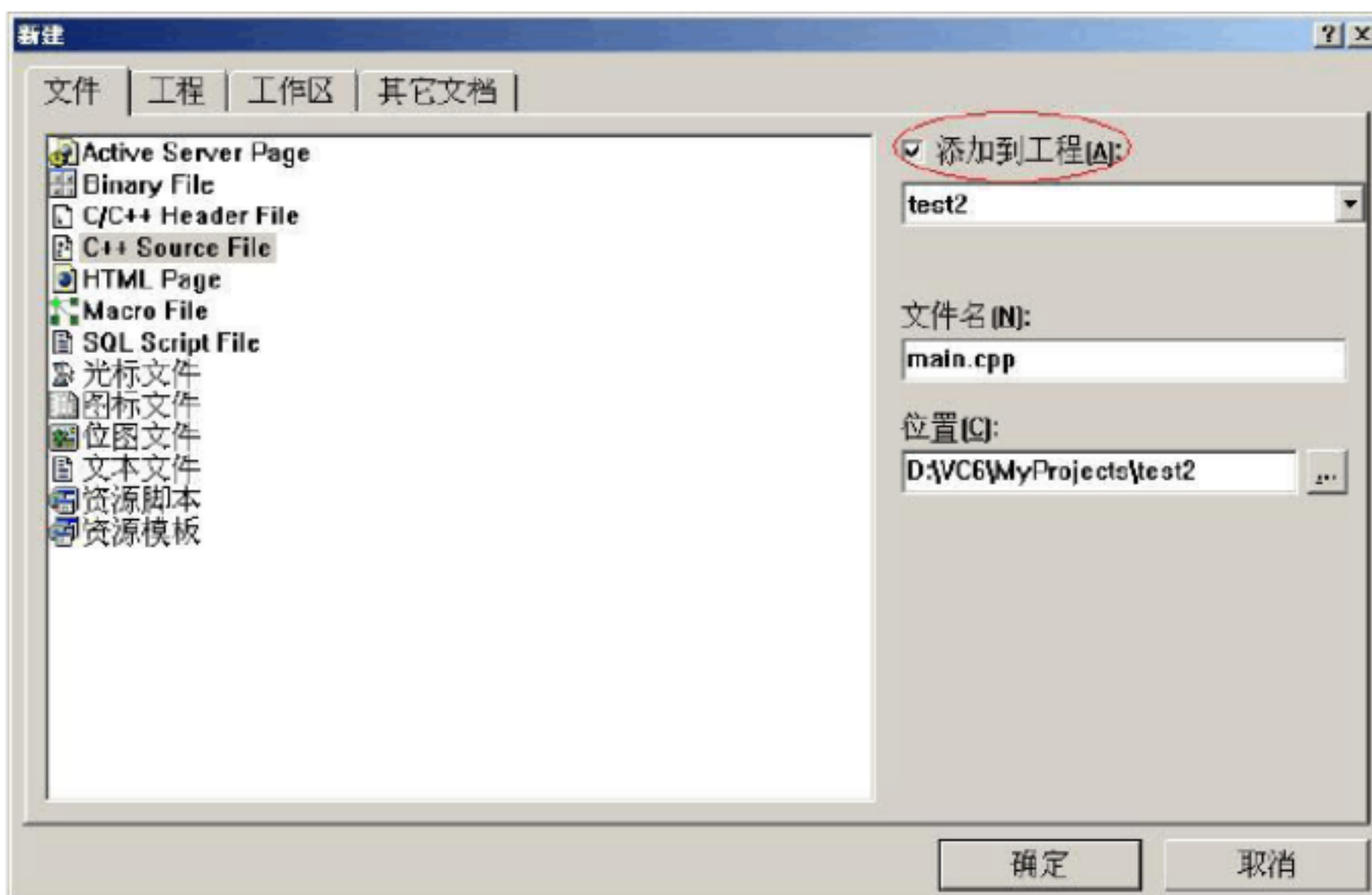
4.1 使用预览

[主页](#) (See 1.)- > 使用预览

使用上，基本和 Turbo C 没太大区别。启动 Visual C++ 6.0，创建一个控制台项目（Win32 Console Application），选择空项目，



然后新建 "C++ Source File" 项，



然后在那个 cpp 里写代码就行了。

看一个画圆的例子吧：

```
#include "graphics.h"           // 就是需要引用这个图形库
int main()
{
    initgraph (640, 480);       // 初始化为 640*480 大小的窗口，这里和 TC 略有区别
    circle (200, 200, 100);      // 画圆，圆心 (200, 200)，半径 100
    getch ();                    // 等待用户按键，按任意键继续
    cegraph ();                  // 关闭图形界面
    return 0;
}
```

呵呵，很简单吧。

不过还是有不少区别的，比如颜色上，TC 只有 16 色，而这个库支持了真彩色。

还有，这个库增加了鼠标、键盘扩展、双缓冲、批量绘图、读取图片（点阵或矢量）等功能。

另外，如果你希望运行时完全不带控制台窗口，如果在 VC 下，默认就是去掉控制台窗口。

但如果你需要显示出来的话，你可以把 `#define SHOW_CONSOLE` 写在 `#include "graphics.h"` 的前面，例如：

```
#define SHOW_CONSOLE
#include "graphics.h"
int main()
{
```

```

    initgraph (640, 480);
    getch ();
    closaph ();
    return 0;
}

```

但，如果你用的是 CFree，那把 main 改 nMain 即可实现控制台窗口的隐藏，如下替换一下即可：

```

#include "graphics.h"
int WinMain // 这里在 graphics.h 里已经定义了宏自动把它展开成合法的声明，
以减免声明
{
    initgraph (640, 480);
    getch ();
    closegraph ();
    return 0;
}

```

当然，你在 VC上也直接这样写也可以，因为也会自动帮你替换回 main，所以直接写 WinMain 可以两边通用。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5. 函数

5.1 库函数目录

[主页 \(See 1.\)](#) - > 库函数目录

目录：

[绘图环境相关函数 \(See 5.4.1\)](#)

[颜色表示及数 \(See 5.2.1\)](#)

[绘制图形相关函数 \(See 5.3.1\)](#)

[文字输出相关函数 \(See 5.5.1\)](#)

[图像处理相关函数 \(See 5.6.1\)](#)

[键盘鼠标输入函数 \(See 5.7.1\)](#)

[时间函数 \(See 5.11.1\)](#)

[数学函数 \(See 5.8.1\)](#)

[随机函数 \(See 5.10.1\)](#)

[其它函数 \(See 5.9.1\)](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2 颜色表示及相关函数

5.2.1 颜色表示及相关函数

函数数据	说明
颜色表示 (See 5.2.2)	介绍颜色表示方法
getbkcolor (See 5.2.7)	获取当前绘图背景色
EGEGET_ (See 5.2.3)	返回指定颜色中的蓝色值
getcolor (See 5.2.8)	获取当前绘图前景色
getfillcolor (See 5.2.9)	获取当前绘图填充色
EGEGET_ (See 5.2.4)	返回指定颜色中的绿色值
EGEGET_ (See 5.2.5)	返回指定颜色中的红色值
hsl2rgb (See 5.2.10)	转换 HSL 颜色为 RGB 颜色
hsv2rgb (See 5.2.11)	转换 HSV 颜色为 RGB 颜色
EGGB (See 5.2.6)	通过红、绿、蓝颜色分量合成颜色
rgb2gray (See 5.2.12)	转换 RGB 颜色为 灰度颜色
rgb2hsl (See 5.2.13)	转换 RGB 颜色为 L 颜色
hsv2rgb (See 5.2.14)	转换 RGB 颜色为 HSV 颜色
setbkcolor (See 5.2.15)	设置当前绘图背景色
setbkcolor_f (See 5.2.16)	设置清屏时所用的背景色
setbkmode (See 5.2.17)	设置输出文字时的背景模式
setcolor (See 5.2.18)	设置当前绘图前景色
setfillcolor (See 5.2.19)	设置当前绘图填充色
setfontbkcolor (See 5.2.20)	设置当前文字背景色

特殊说明：

这里部分函数的最后一个参数为一个 PIMAGE的参数，是一个可选参数，如果不填，则绘画到当前页。如果填上，则设置或者绘画到指定的 IMAGE。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.2 颜色表示

[主页 \(See 1.\)](#)-> [库函数目录 \(See 5.1\)](#)-> [颜色表相关函数 \(See 5.2.1\)](#)-

> 颜色表示

设置绘图色有以下几种办法：

1. 用 16 进制的颜色表示，形式为
0xrrggbb (rr= 红， gg=绿， bb=蓝)

2. 用预定义颜色，如下：

常量值	值	颜色值	常量值	值	颜色
BLACK值	0 值	黑值	DARKGRAY值	0x545454 值	深灰
BLUE值	0x0000A8 值	蓝值	LIGHTBLUE值	0x5454FC 值	亮蓝
GREEN值	0x00A800 值	绿值	LIGHTGREEN值	0x54FC54 值	亮绿
CYAN值	0x00A8A8 值	青值	LIGHTCYAN值	0x54FCFC值	亮青
RED值	0xA80000 值	值	LIGHTRED值	0xFC5454 值	亮红
MAGENTA值	0xA800A8 值	紫值	LIGHTMAGENTA值	0xFC54FC值	亮紫
BROWN值	0xA8A800 值	棕值	YELLOW值	0xFCFC54值	黄
LIGHTGRAY值	0xA8A8A8 值	浅灰值	WHITE 值	0xFCFCFC值	白

3. 用 EGERGB宏合成颜色。详见 EGERGB

4. 用 hsl2rhsv2rgb 转换其他色彩模型到 RGB 颜色。详见 hsl2rgb 、 hsv2rgb 。

示例：

以下是部分设置前景色的方法：

```
setcolor (0xff0000);
setcolor (UE);
setcolor (EGERG(0, 0, 255));
setcolor (hsl2rgb (240, 1, 0.5));
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.3 EGET B

参数：

rgb

指定的颜色。

返回值：

指定颜色中的蓝色值，值的范围 0~255 。

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.4 EGET G

参数：

rgb

指定的颜色。

返回值：

指定颜色中的绿色值，值的范围 0~255 。

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.5 EGET R

参数：

rgb

指定的颜色。

返回值：

指定颜色中的红色值，值的范围 0~255 。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.6 EGERGB

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [颜色表示及相关函数 \(See 5.2.1\)](#)- > EGERGB

功能：

EGERGB宏用于通过红、绿、蓝颜色分量合成颜色。

声明：

```
color_t EGERGB (  
    BYTE byRed,      // 颜色的红色部分  
    BYTE byGreen,    // 颜色的绿色部分  
    BYT byBlue       // 颜色的蓝色部分
```

参数：

byRed
颜色的红色部分，取值范围： 0~255。
byGreen
颜色的绿色部分，取值范围： 0~255。
byBlue
颜色的蓝色部分，取值范围： 0~255。

返回值：

返回合成的颜色。

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.7 getbkcolor

```
color_t  getbkcolor (
    PIMAGEpimg =  NULL
);
```

参数：

（无）

返回值：

返回当前绘图背景色。

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.8 getcolor

```
color_t  getcolor (
    PIMAGEpimg =  NULL
);
```

参数：

（无）

返回值：

返回当前绘图前景色。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.9 getfillcolor

```
color_t getfillcolor (
    PIMAGEpimg = NULL
);
```

参数：

)

返回值：

返回当前绘图填充色。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.10 hsl2rgb

参数：

H

原 HSL 颜色模型的 Hue(色相) 分量, $0 \leq H < 360$ 。

S

原 HSL 颜色模型的 Saturation(饱和度) 分量, $0 \leq S \leq 1$ 。

L

原 HSL 颜色模型的 Lightness(亮度) 分量, $0 \leq L \leq 1$ 。

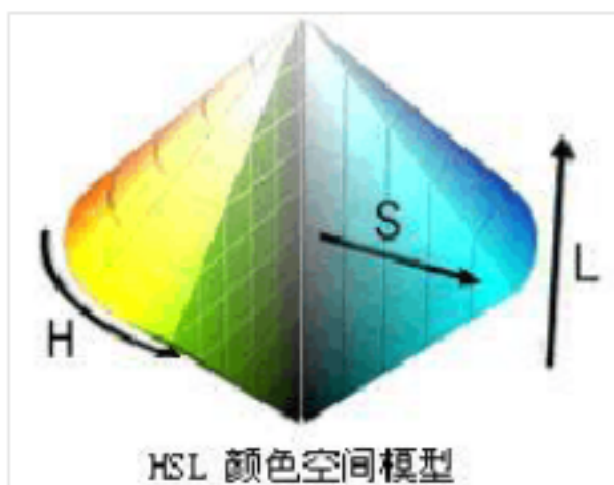
返回值：

对应的 RGB 颜色。

说明：

HSL 又称 HLS。

HSL 的颜色模型如图所示：



H 是英文 Hue 的首字母，表示色相，即组成可见光谱的单色。红色在 0 度，绿色在 120 度，蓝色在 240 度，以此方向过渡。

S 是英文 Saturation 的首字母，表示饱和度，等于 0 时为灰色。在最大饱和度 1 时，具有最纯的色光。

L 是英文 Lightness 的首字母，表示亮度，等于 0 时为黑色，于 0.5 时是色彩最鲜明的状态，等于 1 时为白色。

示例：

请参见示例程序 19 中的“彩虹”。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.11 hsv2rgb

参数：

H

原 HSL 颜色模型的 Hue(色相) 分量, $0 \leq H < 360$

S

原 HSL 颜色模型的 Saturation(饱和度) 分量, $0 \leq S \leq 1$ 。

V

原 HSL 颜色模型的 Value(明度) 分量 , 0 L 1。

返回值：

对应的 RGB 颜色。

说明：

HSV 又称 H。

HSV 的颜色模型如图所示：



H 是英文 Hue 字母，表示色相，即组成可见光谱的单色。 红色在 0 度，绿色在 120 度，蓝色在 240 度，以此方向过渡。

S 是英文 Saturation 首字母，表示饱和度，等于 0 时为灰色。在最大饱和度 1 时，每一色相具有最纯的色光。

V 是英文 Value 的首字母，明度，等于 0 时为黑色，在最大明度 1 时，是色彩最鲜明的状态。

示例：

HSV 颜色模型类似于 HSL，示例程序 19 中的“彩虹”是 HSL 模型的操作范例，可以参考。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.12 rgb2gray

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [颜色表示及相关函数](#) (See 5.2.1)- > rgb2gray

功能：

该函数用于返回与指定颜色对应的灰度值颜色。

声明：

```
color_t  rgb2gray (
    color_t  rgb
);
```

参数：

rgb
原 RGB 颜色。

返回值：

对应的灰度颜色。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.13 rgb2hsl

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [颜色表示及相关函数](#) (See 5.2.1)- > rgb2hsl

功能：

用于转换 RGB 颜色为 HSL 颜色。

声明：

```
void  rgb2hsl (
    color_t  rgb,
    float    * H,
    float    * S,
    float    * L
);
```

参数：

rgb

原 RGB 颜色。

H

HSL 颜色模型的 Hue(色相) 分量, $0 \leq H < 360$ 。

S

原 HSL 颜色模型的 Saturation(饱和度) 分量, $0 \leq S \leq 1$ 。

L

原 HSL 颜色模型的 Lightness(亮度) 分量, $0 \leq L \leq 1$ 。

返回值：

说明：

HSL 详见 hsl2rgb 。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.14 rgb2hsv

[主页 \(See 1.\)](#) - [>](#) [库函数目录 \(See 5.1\)](#) - [>](#) [颜色表示及相关函数 \(See](#)

[5.2.1\)](#) - [>](#) [rhsv](#)

功能：

该函数用于转换 RGB 颜色为 HSV 颜色。

声明：

```
void rgb2hsv (
    color_t rgb,
    float * H,
    float * S,
    float * V
```

);

参数：

rgb

原 RGB 颜色。

H

原 HSL 颜色模型的 Hue(色相) 分量, $0 \leq H < 360$

S

原 HSL 颜色模型的 Saturation(饱和度) 分量, $0 \leq S \leq 1$ 。

V

原 HSL 颜色模型的 Value(明度) 量, $0 \leq V \leq 1$ 。

返回值：

(无)

说明：

HSV 详见 `hsv2rgb`。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.15 setbkcolor

参

color

指定要设置的背景颜色。注意，该设置会同时影响文字背景色。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.16 setbkcolor f

参数：

color
指定要设置的背景颜色。注意，该设置会同时影响文字背景色。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.17 setbkmode

```
void setbkmode (
    t iBkMode,
    PIMAGEpimg = NULL
);
```

参数：

iBkMode
指定输出文字时的背景模式，可以是以下值：

值	描述
OPAQUE值	背景用当前背景色填充（默认）。
TRANSPARENT	背景是透明的。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.18 setcolor

```
void setcolor (
    or_t color,
    PIMAGEpimg = NULL
);
```

参数：

color
要设置的前景颜色。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.19 setfillcolor

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [颜色表示及相关函数](#) (See

5.2.1)- > setcolor

功能：

这个函数用于设置绘图填充色。

声明：

```
void setfillcolor (
    color_t color,
    PIMAGEpimg = NULL
);
```

参数：

color
要设置的填充颜色。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.2.20 setfontbkcolor

```
void setfontbkcolor (
    color_t color,
    PIMAGEpimg = NULL
);
```

参数：

color
要设置的文字背景颜色。

返回值：

(无)

示例：

(

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3 绘制图形相关函数

5.3.1 绘制图形相关函数

函数或数据	说明
arc (See 5.3.2)	画圆弧
bar (See 5.3.3)	画无边框填充矩形
bar3d (See 5.3.4)	画有边框三维填充矩形
circle (See 5.3.5)	画圆
drawbezier (See 5.3.6)	画 bezier 曲线
drawlines (See 5.3.7)	画多条不连续线段
drawpoly (See 5.3.8)	画多边形
ellipse (See 5.3.9)	画线
fillellipse (See 5.3.10)	画填充的椭圆
llpoly (See 5.3.11)	画填充的多边形
floodfill (See 5.3.12)	指定边界色填充区域
floodfillsurface (See 5.3.13)	指定区域色填充区域
getfillcolor (See 5.3.14)	获取当前填充颜色
getfillstyle	获取填充类型（暂不支持）
getheight (See 5.3.15)	获取绘图区的高度
getlinestyle (See 5.3.16)	获取当前线形
getpixel (See 5.3.17)	获取像素点的颜色
getwidth (See 5.3.18)	获取绘图区的宽度
getx (See 5.3.19)	获取当前 x 坐标
gety (See 5.3.20)	获取当前 y 坐标
line (See 5.3.21)	画线
linereI (See 5.3.22)	画线
lineto (See 5.3.23)	画线
moverel (See 5.3.24)	移动当前点

moveto (See 5.3.25)	移动当前点
pieslice (See 5.3.26)	画填充圆扇形
putpixel (See 5.3.27)	画像素点
putpixels (See 5.3.28)	画多个像素点
rectangle (See 5.3.29)	画空心矩形
secl (See 5.3.30)	画填充椭圆扇形
setfillcolor (See 5.3.31)	设置当前填充颜色
setfillstyle (See 5.3.32)	设置当前填充类型（过时函数）
setlinestyle (See 5.3.33)	设置当前线形
setlinewidth (See 5.3.34)	设置当前线宽
setwritemode (See 5.3.35)	设置绘图位操作模式

特殊说明：

以下所有函数的坐标模式为，如果以有向线段表示的量，起点能，终点取不到。比如 line 函数，起点 x1,y1 能画上点，终点 x2,y2 不会画上点，请注意。类似的有 lineto, linerel, bar, b3d, rectangle, ellipse, sector，函数说明内也会对本段进行补充。

另外，这里每一个函数的最后一个参数均为一个 PIMAGE的指针，是一个可选参数，如果不填，则绘画到当前页。如果填上，则绘画到指定的 IMAGE。

关于效率，如果使用手动更新模式（ setrendermode ），那以上所有函数的执行速度都会得到提升。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.2 arc

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)- > arc

功能：

这个函数用于画圆弧。边线颜色由 setcolor 函数决定

声明：

void arc (

```

        int x,
        int y,
        int stangle,
        int endangle,
        int radius,
        PIMAGEpimg = NULL
    );
void arcf (
    float x,
    float y,
    float stangle,
    float endangle,
    float radius,
    PIMAGEpimg = NULL
);

```

参数：

x
圆弧的圆心 x 坐标。

y
圆弧的圆心 y 坐标。

stangle
圆弧的起始角的角度。

endangle
圆弧的终止角的角度。

radius
圆弧的半径。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.3 bar

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [绘制图形相关函数 \(See 5.3.1\)](#)- > bar

功能：

这个函数用于画无边框填充矩形。其中，填充颜色由 `setfillstyle` 函

声明：

```
void bar (
    int left,
    int top,
    int right,
    int bottom,
    PIMAGEpimg = NUL
);
```

参数：

left
矩形左部 x 坐标。

top
矩形上部 y 坐标。

right
矩形右 x 坐标（该点取不到，实际右边界为 right-1 ）。

bottom
矩形下部 y 坐标（该点取不到，实际下边界为 bottom-1 ）。

返回值：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.4 bar3d

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)- > bar3d

:

这个函数用于画有边框三维填充矩形。其中，填充颜色由 `setfillstyle` 函数决定

声明：

```
void bar3d (
    int left,
    int top,
    int right,
    int bottom,
    int depth,
    bool topflag,
    PIMAGEpimg = NULL
);
```

参数：

left

矩形左部 x 坐标。

top

矩形上部 y 坐标

right

矩形右部 x 坐标（该点取不到，实际右边界为 `right-1`）。

bottom

矩形下部 y 坐标（该点取不到，实际为 `bottom-1`）。

depth

矩形深度。

topflag

为 false 时，将不画矩形的三维顶部。该选项可用来画堆叠的三维矩

返回值：

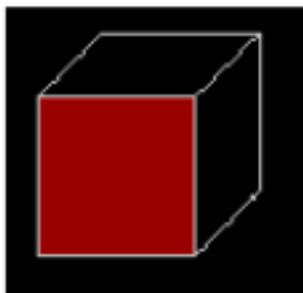
(无)

示例：

```
#include "graphics.h"
```

```
int main()
{
    initgraph (600, 400
    setfillstyle ( RED);
    bar3d (100, 100, 150, 150, 20, 1);
    getch ();
    return 0;
}
```

示例效果：



Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.5 circle

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)

> circle

功能：

这个函数用于画圆。此圆是空心的，不填充，而边线颜色由 seolor 函数决定

声明：

```
void circle (
```

```

        int x,
        t y,
        int ras,
        PIMAGEpimg = NULL
    );
void circlef (
    float x,
    float y,
    float radius,
    PIMAGEpimg = NULL
);

```

参数：

x

圆的圆心 x 坐标。

y

圆的圆心 y 坐标。

radius

圆的半径。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.6 drawbezier

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See

5.3.1)- > drawbezier

功能：

这个函数用于画 bezier 曲线。边线颜色由 setcolor 函数决定

明：

```
void drawbezier (
    int numpoints,
    const int *polypoints,
    PIMAGEpimg = NULL
);
```

参数：

numpoints

多边形点的个数，需要是被 3 除余 1 的数，如果不是，则忽略最后面若干个点。

polypots

每个点的坐标（依次两个分别为 x,y），数组元素个数为 numpoints * 2。

每一条 bezier 曲线由两个端点和两个控制点组成，条则共用端点。

：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.7 drawlines

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See

5.3.1)- > drawlines

功能：

这个函数用于画多条线段。边线颜色由 setcolor 定

声明：


```
void drawlines (
    int numlines,
    const int *polypoints,
    PIMAGEpimg = NULL
);
```

参数：

numlines
线段数目。

polypoints

每个点的坐标（依次两个分别为 x, y ），数组元素个数为 $\text{numlines} * 4$ 。
每两个点画一线段。

返回值：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.8 drawpoly

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See

5.3.1)- > drawpoly

功能：

这个函数用于画多边形。边线颜色由 `setcolor` 函数决定

声明：

```
void drawpoly
    int numlines,
    const int *polypoints,
    PIMAGEpimg = NULL
```

);

参数：

numpoints

多边形点的个数。

polypoints

每个点的坐标（依次两个分别为 x, y ），数组元素个数为 $\text{numpos} * 2$ 。
该函数并不会自动连接多边形首尾。如果需要画封闭的多边形，请将最后一个点设置为与第一点相同。

返回值：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.9 ellipse

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)- ellipse

功能：

这个函数用于画椭圆弧线。边线颜色由 `setcolor` 函数决定

声明：

```
void ellipse (
    int x,
    int y,
    int stangle,
    int endangle,
    int xradius,
    int yradius,
```

```

        PIMAGEpimg = NULL
    );
void ellipsef (
    float x,
    float y,
    float stangle,
    float endangle,
    float xradius,
    float yradius,
    PIMAGEpimg = NULL
);

```

参数：

x
椭圆弧线的圆心 x 坐标。

y
椭圆弧线的圆心 y 坐标

stangle
椭圆弧线的起始角的角度。

endangle
椭圆弧线的终止角的角度。

xradius
椭圆弧线的 x 轴半径。

yradius
椭圆弧线的 y 轴半径。

返回值：

（无）

示例：

（

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.10 fillellipse

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)- > fillellipse

功能：

这个函数用于画填充的椭圆。边线颜色由 `setcolor` 函数决定，填充颜色由 `setfillstyle` 函数决定。

声明：

```
void fillellipse (
    int x,
    int y,
    int xradius,
    int yradius,
    PIMAGEpimg = NULL
);

void fillellipsef (
    float x,
    float y,
    float xradius,
    float yradius,
    PIMAGEpimg = NULL
);
```

参数：

x
椭圆的圆心 x 坐标。

y
圆的圆心 y 坐标。

xradius
椭圆的 x 轴半径。

yradius
椭圆的 y 轴半径。

返回值：

(无)

示例

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.11 fillpoly

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [绘制图形相关函数 \(See 5.3.1\)](#)- > fillpoly

功能：

这个函数画填充的多边形。边线颜色由 `setcolor` 函数决定，填充颜色由 `setfillstyle` 函数决定

声明：

```
void fillpoly (
    int numpoints,
    const int *polypoints,
    PIMAGEpimg = NULL
);
```

参数：

numpoints
多边形点的个数。

polypoints
每个点的坐标，数组元素个数为 `numpoints * 2`。
该函数会自动连接多边形首尾。

返回值：

(无)

示例：

(无)

说明：

如果这个多边形发生自相交，那么自交次数为奇数的区域则不填充，偶数次的填充，不自交就是偶数次。不过这样说明相信非常难理解，个例子：

```
#include "graphics.h"
int main()
{
    initaph (600, 400);
    setfillsty ( RED);
    int pt[] = {
        0, 0,
        0, 0,
        100, 100,
        10, 10,
        90, 10,
        0, 100,
    };
    fillpoly (6, pt);
    getch ();
    return 0;
}
```

运行结果：



第二个

```
#include "graphics.h"

int mr()
{
    initgraph (600, 400);
    setfillstyle ( RED);
    int pt[] = {
        0, 0,
        100, 0,
```

```

    100, 100,
    0, 100,
    0, 0,
    100, 0,
    100, 120,
    0, 100,
};
llpoly (8, pt);
getch ();
return 0;
}

```

运行结果：



Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.12 floodfill

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘相关函数](#) (See 5.3.1)- >

floodfill

功能：

这个函数使用 `sillstyle` 设置的填充方式对区域进行填充。填充颜色由 `setfillstyle` 函数决定。

声明：

```

void floodfill (
    int x,
    int y,
    int border,
    PIMAGEpimg = NU
);

```


参数：

x

填充的起始点 x 坐标。

y

填充的起始点 y 坐标。

border

填充的边界颜色。填充动作在该颜色围成的区域内填充。如果该颜色围成的区域不封闭，那么将使全屏幕都填充上。

返回值：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.13 floodfill

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [绘制图形相关函数 \(See 5.3.1\)](#)- > floodfillsurface

功能：

这个函数使用 `setfillstyle` 设置的填充方式对区域进行填充。填充颜色由 `setfillst` 函数决定。

声明：

```
void floodfillsurface (
    int x,
    int y,
    color_t areacolor,
    PIMAGEpimg = NULL
```

);

参数：

x

填充的起始点 x 坐标。

y

填充的起始点 y 坐标。

areacolor

填充区域的原本等于该颜色且与起始点相连所构成的区域，由 setfillcolor 所设置的颜色来填充。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.14 getfillcolor

```
COLORREF getfillcolor (
    IMAGE* p = NULL
);
```

参数：

(无)

返回值：

当前的填充颜色

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.15 getheight

```
int  getheight (
    PIMAGEpimg =  NULL
);
```

参数：

(无)

返回值：

返回图片高度。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.16 getlinestyle

```
void  getlinestyle (
    int    *plinestyle,
    WORD*pattern =  NULL,
    int    *pthickness =  NL,
    PIMAGEpi =  NULL
);
```

参数：

plinestyle

返回当前线型。详见 `setlinestyle` 。

pattern

返回当前自定义线形数据。

pthickness
返回当前线形宽度。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.17 getpixel

```

COLORREF getpixel(
    int x,
    int y
    MAGPimg = NULL
);
    
```

参数：

x
要获取颜色的 x 坐标。

y
要获取颜色的 y 坐标。

返回值：

指定点的颜色。

示例：

(无)

其它说明：另有高速版的 `getpixel_f` 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方程序结果莫名其妙，甚至直接崩溃），并且必须在批量绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.18 getwidth

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [绘制图形相关函数 \(See 5.3.1\)](#)- > getwidth

功能：

这个函数用于获取图片宽度。

声明：

```
int getwidth (
    PIMAGEpimg = NULL
);
```

参数：

(无)

返回值：

返回图片宽度。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.19 getx

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#) [绘制图形相关函数 \(See 5.3.1\)](#)- > getx

功能：

这个函数用于获取当前 x 坐标。

声明：

```
int  getx (
    PIMAGEpimg =  NULL
);
```

参数：

(无)

值：

返回当前 x 坐标。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.20 gety

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘制图形相关函数](#) (See

5.3.1)- > gety

功能：

这个函数用当前 y 坐标。

：

```
int  gety (
    PIMAGEpimg =  NULL
);
```

参数：

(无)

返回值

返回当前 y 坐标。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.21 line

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [绘制图形相关函数 \(See 5.3.1\)](#)- > line

功能：

这个函数用于画线。

声明：

```
void line (
    int x1,
    int y1,
    int x2,
    int y2,
    PIMAGEpimg = NULL
);
```

数：

x1

线的起始点的 x 坐标。

y1

线的起始点的 y 坐标。

x

线的终止点的 x 坐标（该点本身画不到）。

y2

线的终止点的 y 坐标（该点本身画不到）。

返回值：

)

示例：

（无）

其它说明：另有高速版的 `line_f` 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方，要么程序结果莫名其妙，甚至直接崩溃），并且必须在窗口锁定绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.22 linerel

```
void linerel (
    int dx,
    int dy,
    PIMAGEpimg = NULL
);
```

参数：

dx

从“当前点” cx 开始画线，x 轴偏移 dx，终点为 cx+dx（终点本身画不到）。

dy

从“当前点” cy 开始画线，沿 y 轴偏移 dy，终点为 cy+dy（终点本到）。

返回值：

（无）

示例：

（无）

其它说明：另有高速版的 `linere1_f` 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方，要么程序结果莫名其妙，甚至直接崩溃），并且必须在窗口锁定绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.23 lineto

```
void lineto (
    in x,
    int y,
    PIMAGEpimg = NULL
);
```

参数

x

从“当前点”开始画线，终点横坐标为 x （终点本身画不到）。

y

从“当前点”开始画线，终点纵坐标为 y （终点本身画不到）。

返回值：

（无）

示例：

（无）

说明：另有高速版的 `lineto_f` 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方，要么程序结果莫名其妙，甚至崩溃），并且必须在窗口锁定绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.24 moverel

参数：

dx

将当前点沿 x 轴移动 dx 。

dy

将当前点沿 y 轴移动 dy 。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.25 moveto

参数：

x

新的当前点 x 坐标。

y

新的当前点 y 坐标。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.26 pieslice

```
void pieslice (
    int x,
    int y,
    int stangle,
    int endangle,
    int radius,
    PIMAGEpg = NULL
);

void pieslicef (
    float x,
    float y,
    float stangle,
    float endangle,
    float radius,
    PIMAGEpi = NULL
);
```

参数：

x
圆扇形的圆心 x 坐标。

y
圆扇形的圆心 y 坐标。

stangle
圆扇形的起始角的角度。

endangle
圆扇形的终止角的角度。

radius
圆扇形的半径。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.27 putpixel

```
void putpixel (
    int x,
    int y,
    COLORREF color,
    PIMAGEimg = NULL
);
```

参数：

x
点的 x 坐标。

y
点的 y 坐标。

color
点的颜色。

返回值：

(无)

示例：

(无)

说明：另有高速版的 `putpixel_f` 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方，要么程序结果莫名其妙，甚崩溃），并且必须在窗口锁定绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.28 putpixels

```
void putpixels (
    int nPoint,
    int * pPoints,
    PIMAGEpimg = NULL
);
```

参数：

nPoint
点的数目。

pPoints
指向点的描述的指针，一个 int 型的数组，依次每三个 int 描述一个点：第 x 坐标，第二个为 y 坐标，第三个为颜色值。

返回值：

(无)

示例：

(无)

其它说明：另有高速版的 putpixels_f 函数，参数一样，作用一样，但不进行相对坐标变换和边界检查（如果越界绘图，要么画错地方，要么程序结果莫名其妙，甚至直接崩溃），并且必须在窗口锁定绘图模式下才能使用，否则将发生不可预知的结果。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.29 rectangle

```
void rectangle (
    int left,
    int top,
    int right,
    int bottom,
    PGEpimg = NULL
);
```

参数：

left

矩形左部 x 坐标。

top

矩形上部 y 坐标。

right

矩形右部 x 坐标。

bom

矩形下部 y 坐标。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.30 sector

```
void sector (
    int x,
    int y,
    int stangle,
    int endangle,
    int xradius,
    int yradius,
    PIMAGEpimg = NULL
);

void ectorf (
    float x,
    float y,
    float stangle,
    float endangle,
    float xradius,
```



```
float    yradius,  
PIMAGEpi =    NULL  
);
```

参数：

x
椭圆扇形的圆心 x 坐标。

y
椭圆扇形的圆心 y 坐标。

stangle
椭圆扇形的起始角的角度。

endangle
椭圆扇形的终止角的角度。

xradius
椭圆扇形的 x 轴半径。

yradius
椭圆扇形的 y 轴半径。

返回值：

（无）

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.31 setfillcolor

```
void setfillcolor (   
    COLORREFcolor,  
    PIMAGEpimg =    NULL  
);
```

参数：

color
填充颜色。

返回值：

(无)

示例：

设置蓝色固实填充：
setfillcolor(BLUE);

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.32 setfillstyle

[主页](#) (See 1.)- > [库函数目](#) (See 5.1)- > [绘制图形相关函数](#) (See 5.3.1)-
> setfillstyle

功能：

这个函数用于设置当前填充类型。该函数的自定义填充部分尚不支持。

声明：

```
void setfillstyle (
    int pattern,
    COLORREF color,
    PIMAGE pimg = NULL
);
```

参数：

pattern
填充类型，可以是以下宏或值：

宏	值	含义
NULL_FILL	1	不填充

SOLID_FILL	2	固实填充
------------	---	------

pupattern
color
填充颜色。

指定图案填充时的样式，目前无作用。

返回值：
(无)

示例：
设置蓝色固实填充：
setfillstyle(SOLID_FILL, BLUE);

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.33 setlinestyle

```
void setlinestyle (
    int linestyle,
    WORD pattern = NULL,
    int thickness = 1,
    PIMAGE pimg = NULL
);
```

参数：
linestyle
线型，可以是以下值：

	含义
SOLID_LINE	线形为实线。
CENTER_LINE	线形为：- - - - -
DOTTED_LINE	线形为：
DASHED_LINE	线形为：- - - - -
NULL_LINE	线形为不可见。

USERBIT_LINE	线形样式是自定义的，依赖于 upattern 参数。
--------------	----------------------------

upattern

自定义线形数据。

自定义规则：该数据为 WORD 类型，共 16 个二进制位，每位为 1 表示画线，为 0 示空白。从低位到高位表示从起始到终止的方向。

仅当线型为 PS_USERSTYLE 时该参数有效。

thickness

线形宽度。

返回值：

(无)

示例：

设置线形为点划线： setlinestyle(PS_DASHDOT);

设置线形为宽度 3 像素的虚线： setlinestyle(PS_DASH, NULL, 3);

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.34 setlinewidth

```
void setlinewidth (
    int thickness,
    PIMAGE pimg = NULL
);
```

参数：

thicss

线形宽度。

回值：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.3.35 setwritemode

```
void setwritemode
    int mode.
    PIMAGEpimg = NULL
);
```

参数：

mode
二元光栅操作码（即位操作模式），支持全部的 16 种二元光栅操作码，罗列如下：

位操作模式	描述
R2_BLACK	绘制像素颜色 = 黑色
R2_COPYPEN	绘制出的像素颜色 = 当前颜色（默认）
R2_MASKNOTPEN	绘制出的像素颜色 = 屏幕颜色 AND (NOT 当前颜色)
2_MASKPEN	绘制出的像素颜色 = 屏幕颜色 AN 前颜色
R2_MASKPENNOT	绘制出的像素颜色 = (NOT 屏幕颜色) AND 当前颜色
R2_MERGEOTPEN	绘制出的像素颜色 = 屏幕颜色 OR (NOT 当前颜色)
R2_MERGEPEN	绘制素颜色 = 屏幕颜色 OR 当前颜色
R2_MERGEPENNT	绘制出的像素颜色 = (NOT 屏幕颜色) 当前颜色
R2_NOP	绘制出的像素颜色 = 屏幕颜色
R2_NOT	绘制出的像素颜色 = NOT 屏幕颜色
R2_NOTCOPYPEN	绘制出的像素颜色 = NOT 当前颜色
R2_NOTMASKPEN	绘制出的像素颜色 = NOT (屏幕颜色 AND 当前颜色)
R2_NOTMERGEPEN	绘制出的像素颜色 = NOT (屏幕颜色 OR 当前颜色)
R2_NOTXORPEN	绘制出的像素颜色 = NOT (屏幕颜色 XOR 当前颜色)
R2_WHITE	绘制出的像素颜色 = 白色
R2_XORPEN	绘制出的像素颜色 = 屏幕颜色 XOR 当前颜色

注：1. AND / OR / NOT / XOR 为布尔位运算。 2. " 屏幕颜色 " 指绘制所经过的屏幕像素点的颜色。 3. " 当前颜色 " 是指通过 tcolor 设置的用于当前绘制的颜色。

返回值：

（无）

示例：
(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4 绘图环境相关函数

5.4.1 绘图环境相关函数

函数或数据	
cleardevice (See 5.4.2)	清除屏幕
clearviewport (See 5.4.3)	清空视图
closegraph (See 5.4.4)	关闭图形环境
ettarget (See 5.4.5)	获取当前绘图对象
getviewport (See 5.4.6)	获取当前视图信息
initgraph .(See 5.4.7)	初始化绘图环境
is_run (See 5.4.8)	判断窗口环境是否还在
setactivepage (See 5.4.9)	设置当前绘图页
setcaption (See 5.4.10)	设置窗口标题
setinitmode (See 5.4.11)	设置初始化参数
setrendermode (See 5.4.12)	设置窗口更新模式
settarget (See 5.4.13)	设置当前绘图对象
setviewport (See 5.4.14)	设置当前视图
setvisualpage (See 5.4.15)	设置显示页，把页面内容输出到窗口的页
window_getewport (See 5.4.16)	获取当前窗口可见部
window_setviewport (See 5.4.17)	设置窗口可见部分

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.2 cleardevice

参数

pimg

指定要清除的 PIMAGE, 可选参数。如果不填本参数, 则清除屏幕。

返回值：

(无)

示例：

```
#include "graphics.h"
int main()
{
    initgraph (640, 480);
    circle (200, 200, 100);
    getch ();
    cleardevice ();
    getch ();
    closegraph ();
    return 0;
}
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.3 clearviewport

参数：

pimg

见 setviewport

返回值：

(无)

示

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.4 closegraph

void closegraph ();

参数：

(无)

返回值：

(无)

示例：

```
#include "graphics.h"
int main()

    initgraph (640, 480);
    circle (200, 200, 100);
    getch ();
    cleavice ();
    getch ();
    closegraph ();
    return 0;
}
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.5 gettarget

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [绘图环境相关函数](#) (See

5.4.1)- > gettarget

功能：

这个函数用于获取当前绘图对象。

声明：

PIMAGE gettarget ();

参数：（无） **返回值：** PIMAGE对象 **示例：**（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.6 getviewport

```
void getviewport (  
    int *pleft,  
    int *ptop,  
    int *pright,  
    int *pboom,  
    int *pclip = NULL,  
    PIMAGEpimg = NULL  
);
```

参

pleft

返回当前视图的左部 x 坐标。

ptop

返回当前视图的上部 y

pright

返回当前视图的右部 x 坐标。

pbottom

返回当前视图的下部 y 坐标。

pclip

返回当前视图标志。

pimg

详见 setviewport 的说明

返回值：

（无）

示例：

参见 setviewport

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.7 initgraph

```
void initgraph (
    int Width,
    int Hht,
    int Flag = INIT_DEFAULT
);
void initgraph (
    int * gdriver,
    int * gmode,
    char * path
); // 兼容 Borland C++ 3.1 的重载，不建议使用。
```

参数：

Width

绘图环境的宽度。如果为 -1，则使用屏幕的宽度

Height

绘图环境的高度。如果为 -1，屏幕的高度

Style

请留空，为保留参数

返回值：

(无)

示例：

```
#include "graphics.h"
int main()
{
    initgraph (640, 80);
    circle (200, 200, 10;
    getch ();
    cleardevice ();
    getch ();
    closegraph ();
    return 0;
```

}

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.8 is_run

```
void is_run ();
```

参数：

(

返回值：

0 表示窗口被关闭了

1 表示窗口没有被关闭，程序还在运行

示例：

详见入门示例里的动画部分

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.9 setactivepage

```
void etactivepage (int page);
```

参数：

page

绘图页，范围从，范围从，越界会导致程序错误。默认值为 0

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by

TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.10 setcaption

```
void seption (LPCSTR caon);
void setcaption (LPCWSTRcaption);
```

参数：

caption

指定要设置的标题字符串，可用宽字符或 MBCS字符串，需要使用 gbk 编码。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.11 setinitmode

```
void setinitmode (in mode, int x = CW_USEDEFAULT, int y = CW_USEDEFAULT)
```

参数：

mode

初始化模式，是二进制组合的值。如果为 INIT_DEFAULT 表示使用默认值。

可以使用的值的组合：

INIT_NOBORDER为无边框窗口

INIT_CHILD 为子窗口（需要使用 attachHWND 指定要依附的父窗口，此函数不另说明）

INIT_PMOST 使窗口总在最前

INIT_RENDERMANUA手动更新标志，即调用 delay_fps/delay_ms 等会等待操作的函数时会更新否则保持窗口内容

INIT_WITHLOGO使 initgraph 的时候显示开场动画 logo

INIT_NOFORCEEXIT使关闭窗口的时候退出程序，但窗口会消失，需要配合 is_run 函数

INIT_DEFAULT 默认参数，不调用本函数时即使用此参数

INIT_ANIMATION INIT_DEFAULT, INIT_RENDERMANUAL, INIT_NOFORCEEXIT的组合，用于动画编写

x, y

初始化时窗上角在屏幕的坐标，默认为系统分配。

返回值：

(无)

说明：

本函数只能在 initgraph 前调用。

示例

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.12 setrendermode

参数：

mode

值可能为 RENDER_AUTO或者 RENDER_MANUAL前者自动（默认值），后者手动
自动模式用于简单绘图，手动模式用于制作动画或者游戏

所谓手动模式，即需要调用 delay_fps/delay_ms 等带有等待特性的函数时才会更新窗口

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.13 settarget

void settarget (PIMAGE pbuf = NULL);

参数：

pbuf

绘图对象，为 PIMAGE类型，你要绘画到的 PIMAGE, 如果不填，则还原为窗口

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.14 setviewport

[主页 \(See 1.\)](#) - > [库函数目录 \(See 5.1\)](#) > [绘图环境相关函数 \(See 5.4.1\)](#) -

> setviewport

功能：

这个函数用于设置当前视图。并且，将坐标原点移动到新的视图的 (0, 0) 位置。

声明：

```
void setviewport (
    int left,
    int top,
    int right,
    int bottom,
    int clip = 1,
    PIMAGEpimg = NULL
);
```

参数：

left

视左部 x 坐标。

top

视图的上部 y 坐标。 (left, top) 将成为新的原点。

right

视图的右部 x 坐标。

bot

视图的下部 y 坐标。 (right-1, bottom-1) 是视图的右下角坐标。

clip

视图的裁剪标志。如果为真，所有超出视图区域的会被裁剪掉。

pimg

要设置的图片

返回值：

(无)

示例：

```
#include "graphics.h"
int main()
{
    initgraph (640);
    setviewport (100, 1, 200, 200);
    rectangle (0, 0, 200, 200);
    getch ();
    closegraph ();
    return 0;
}
```

注意：

端点和下端点取不到，上端点和左端点能取到。另外，这函数的最后一个参数为一个 IMAGE*的指针，是一个可选参数，如果不填，则设置到当前页。如果填上，则指定的 IMAGE。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.15 setvisualpage

参数：

page

绘图页，范围从，范围从 0-3，越界会导致程序错误。默认值为 0

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.16 window_getviewport

```
void window_getviewport (  
    int *pleft,  
    int *ptop,  
    int *pright,  
    int *pbottom,  
);
```

参数：

pleft

返回当前视图的左部 x 坐标。

ptop

返回当前视图的上部 y 坐标。

pright

返回当前视图的右部 x 坐标。

pbottom

返回当前视图的下部 y 坐标。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.4.17 window_setviewport

```
vo window_setviewport (
    int left,
    int top,
    int right,
    int bottom
);
```

参数：

left

可见区域的左部 x 坐标。

top

可见区域的上部 y 坐标。 (left, top) 将成为新的原点。

right

可见区域的右部 x 坐标。

bottom

可见区域的下部 y 坐标。 (right-1, bottom-1) 是视图角坐标。

返回值

(无)

示例：

```
#include "graphics.h"
int main()
{
    initgraph (640, 480);
    window_setviewport (100, 100, 400, 400);
    rectang (0, 0, 200, 200);
    getch ();
    closegraph ();
    return 0;
}
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5 文字输出相关函数

5.5.1 文字输出相关函数

函数或数据	说明
getfont (See 5.5.2)	获取当前字体样式
LOGFONT 结构体 (See 5.5.3)	保存字体样式的结构体
outtext (See 5.5.4)	在当前位置输出字符串
outtextrect (See 5.5.5)	在指定矩形区域内输出字符串
outtex (See 5.5.6)	在指定位置输出字符串
rectprintf (See 5.5.11)	在指定矩形格式化输出字符串
seont (See 5.5.7)	设置当前字体样式
settextjustify (See 5.5.8)	设置当前文字对齐方式
textheight (See 5.5.9)	获取字符串的高
textwidth (See 5.5.10)	获取字符串的宽
xyprintf (See 5.5.12)	在指定位置格式化输出字符串

特殊说明：

这里部分函数的最后一个参数为一个 PIMAGE的参数，是一个可选参数，如果不填，则绘画到当前页。如果填上，则设置或者绘画到指定的 IMAGE。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.2 getfont

```
void getfont (
    LOGFONT*font
    PIMAGEpimg = NULL
);
```

参数：

font
指向 LOGFONT 结构体的指针。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.3 logfont

```

struct LOGFONT{
    LONGIfHei;
    LONGIfWidth
    LONGIfEscapement;
    LONGIfOrientation;
    LONGIfWeight;
    BYTE IfItalic;
    BYTE IfUnderline;
    BYTE IfStrikeOut;
    BYTE IfCharSet;
    BYTE IfOutPision;
    BYT IfClipPrecision;
    BYTE IfQuality;
    BYTE IfchAndFamily;
    TCHARIfFaceName[ LF_FACESIZE];
};

```

成员

IfHeight

指定高度（逻辑单位）。

如果为正，表示指定的高度是字体的度；如果为负，表示指定的高度不包含 tmInternalLeading 的高度。也就是说相同绝对值下， 负的比正的会稍高一些； 而实际输出的字体高度，正数时配，负数时，和实际略有偏差，绝对值比实际的略小。

IfWidth

指定字符的平均宽度（逻辑单位）。如果为 0，则比例自适应。

IfEscapement

字符串的书写角度，单位 0.1 度，默认为 0。

IfOrientation

每个字符的书写角度，单位 0.1 度，默认为 0 。

IfWeight

字符的笔画粗细，范围 0~1000 ，0 表示默认粗细，使用数字或下表中定义的宏均可。

宏	粗
FW_DONTCARE	0
FW_THIN	100
FW_EXTRALIGHT	200
FW_ULTRALIGHT	200
FW_LIGHT	300
FW_NORMAL	400
FW_REGULAR	400
FWEDIUM	500
FW_SEMIBOLD	600
FW_DEMIBOLD	600
FW_BOLD	700
FW_EXTRABOLD	800
FW_ULTRABOLD	800
FW_HEAVY	900
FW_BLACK	900

IfIta

指定字体是否是斜体。

IfUnderline

指定字体是否有下划线。

IfStrikeOut

指定字体是否有删除线。

IfChar

指定字符集。以下是预定义的值：

ANSI_CHARSET

BALTIC_CHARSET

CHINESEBIG5_CHARSET

DAULT_CHARSET

EASTEUROPE_CHARSET

GB2312_CHARSET

GREEK_CHARSET

HANGUL_CHET

MAC_CHARSET

OEM_CHARSET

RUSSIAN_CHARSET
SHIFTJIS_CHARSET
SYMBOL_CHARSET
TURKISH_CHARSET

其中，OEM_CHARSET表示字符集依赖本地操作系统。
DEFAULT_CHARSET表示字符集基于本地操作系统。例如，系统位置是 English (United States), 字符集将设置为 ANSI_CHARSET

IfOutPrecision

指定文字的输出精度。输出精度定义输出与所请求的字体高度、宽度、字符方向、行距、间距和字体类型相匹配必须达到的匹配程度。可以是以下值：

值	含义
OUT_DEFAULT_PRECIS	指定默认的映射行为。
OUT_DICE_PRECIS	当系统包含多个名称相同的字体时，备字体。
OUT_OUTLINE_PRECIS	指定字体映射选择 TrueType 和其它的 outline-based 字体。
OUT_RASTER_PRECIS	当包含多个名称相同的字体时，指定光栅字体（即点阵字体）。
OUT_STRING_PRECIS	这个值并不能用于指定字体映射，只是指定点阵字体枚举数据。
OUT_STROKE_PRECIS	这个值并不能用于指定字体映射，只是指定 TrueType 和其他的 outline-based 字体，以及矢量字体的枚举数据。
OUT_TT_ONLY_PRECIS	指定字体映射只选择 TrueType 字体。如果系统中没有安装 TrueType 字体，将选择默认操作。
OUT_TT_PRECIS	当系统包含多个名称相同的字体时，指定 TrueType 字体。

IfClipPrecision

指定文字的剪辑精度。剪辑精度定义如何剪辑字符的一部分位于剪辑区域之外的字符。可以是以下值：

值	含义
CLIP_DELT_PRECIS	指定默认的剪辑行为。
CLIP_STROKE_PRECIS	这个值并不能用于指定字体映射，只是指定光栅（即点阵）、矢量或 TrueType 字体的枚举数据。
CLIP_EMBEDDED	当使用内嵌的只读字体时，必须指个标志。
CLIP_LH_ANGLES	如果指定了该值，所有字体的旋转都依赖于坐标系统的方向是逆时针或顺时针。如果没有指定该值，设备字体始终逆时针旋转，但是其它字体的旋转依赖于坐标系统向。该设置影响 IfOrientation 参数的效果。

IfQuality

指定文字的输出质。输出质量定义图形设备界面（GDI）必须尝试将逻辑字体属性与实际物理字体的字体属性进行匹配的仔细程度。可以是以下值：

值	含义
---	----

ANTIALIASED_QUALITY	指定输出质量是抗锯齿的（如果字体支持）。
DEFAULT_QUALITY	指定输出质量不重要。
DRAFT_QUALITY	草稿质量。字体的显示质量是不重要的。对于光栅字体（即点阵字体），是有效的，这就意味着可以使用更多的尺寸，但是显示质量并不高。如果需要，粗体、斜体、下划线和删除线字体会被合成。
NONANTIALIASED_QUALITY	指定输出质量不是抗锯齿的。
PROOF_QUALITY	正稿质量。指定字体质量比匹配字体属性更重要。对于光栅字体（即点阵字体），缩放是无效的，会选用其最接近的字体大小。虽然选中 PROOF_QUALITY时字体大小不能精确地映射，但是输出质量很高，并且不会有畸变现象。如果，粗体、斜体、下划线和删除线字体会被合成。

如果 ANTIALIASED_QUALITY 和 NONANTIALIASED_QUALITY 都未被选择，抗锯齿效果将依赖于控制面板中字体抗锯齿的设置。

IfPitchAndFamily

指定以常规方式体的字体系列。字体系列描述大致的字体外观。字体系列用于在所需精确字体不可用时指定字体。

1~2 位指定字体间距，可以是以下值：

值	含义
DEFAULT_PITCH	指定默认间距。
FIXED_PITCH	定间距。
VARIABLE_PITCH	指定可变间距。

4~7 位指定字体系列，可以是以下值：

值	含义
FF_DECOTIVE	指定特殊字体。例如 Old English。
FF_DONTCARE	指定字体系列不重要。
FF_MODERN	指定具有或不具有衬线的等宽字体。例如， Pica、Elite 和 Courier New 都是等宽字体。
F_ROMAN	指定具有衬线的等比字体。例如 MS Seri
FF_SCRIPT	指定设计为类似手写体的字体。例如 Script 和 Cursive。
FF_SWISS	指定不具有衬线的等比字体。例如 MS Sans Serif

字体间距和字体系列可以用布尔运算符 OR 连接（即符号 |）。

IfFaceName

字体名称，名称不得超过 31 个字符。如果是空字符串，系统将使用第一个满足其它属性的字体。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.4 outtext

```
void outtext (
    LPCSTR textstring,
    PIMAGEpimg = NULL
);
void outtext (
    HARC,
    PIMAGEpimg = NULL
);
void outtext (
    LSTR textstring,
    PIMAGEpimg = NULL
);
void text (
    WCHAR,
    PIMAGEpimg = NULL
);
```

参数：

textstring
要输出的字符串的指针。

c
输出的字符。

返回

(无)

示例：

```
// 输出字符串
char s[] = "Hello World" ;
outtext (s);
// 输出字符
char c = 'A' ;
outtext (c);
// 输出数值，先将数字格式化输出为字符串
```



```
char s[5];  
sprintf (s, "%d", 1024);  
outtext (s);
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.5 outtextrect

```
void outtextrect (  
    int x,  
    int y,  
    int w,  
    int h,  
    LPCSTRtextstring,  
    PIMAGEpimg = NULL  
);  
void outtextrect (  
    int x,  
    int y,  
    int w,  
    int h,  
    LPCWSTRtextstring,  
    PIMAGEpimg = NULL  
);
```

参数：

x, y, w, h

要输出字符串所在的矩形区域

textstring

要输出的字符串的指针。

返回值：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.6 outtextxy

```
void outtextxy (
    int x,
    int y,
    LPCSTR textstring,
    PIMAGE pimg = NULL
);

void outtextxy (
    int x,
    int y,
    CHAR c,
    PIMAGE pimg = NULL
);

void outtextxy (
    int x,
    int y,
    LPCWSTR textstring,
    PIMAGE pimg = NULL
);

void outwtextxy (
    int x,
    int y,
    WCHAR c,
    PIMAGE pimg = NULL
);
```

参数：

x
字符串输出时头字母的 x 轴的坐标值

y
字符串输出时头字母的 y 的坐标值。

textstring
要输出的字符串的指针。

c
要输出的字符。

返回值：

(无)

示例：

```
// 输出字符串
char s[] = "Hello World" ;
ttextxy (10, 20, s);
// 输出字符
char c = 'A' ;
outtey (10, 40, c);
// 输出数值，先将数字格式化输出为字符串
char s[5];
sprintf (s, "%d", 1024);
outtext (10, 60, s);
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.7 setfont

```
void setfont (
    int nHeight,
    int nWidth,
    LPCSTRlpzFace,
    PIMAGEpimg = NULL
);
void setfont (
    int nHeight,
    int nWidth,
    LPCSTRlpzFace,
    int nEscapement,
    int nOrientation,
    int nWeight,
    bool bItalic,
    bool bUnderline,
    bool bStrikeOut,
    PIMAGEpimg = NULL
);
void setfont (
    int nHeight,
    int nWidth,
    LPCSTRlpzFace,
    int nEscapent,
    int nOrientation,
    int nWeight,
    bool bItalic,
    bool bUnderline,
```

```
bool bStrikeOut,
BYTE fbCharSet,
BYTE fbOutPision,
BYTE fbClipPrecisio
BYTE fbQuality,
BYTE fbPitchAndFamily,
PIMAGEpimg = NULL
);

void setfont (
const LOGFONT*font,
PIMAGEpimg = NULL
);
```

参数：

nHeight
指定高度（逻辑单位）。如果为正，表示指定的高度包括字认行距；如果为负，表示指定的高度只是字符的高度。

nWidth
字符的平均宽度（逻辑单位）。如果为 0，则比例自适应。

lpszFace
名称。对于此参数均有 LPCSTR和 LPCWSTR两个版本，以上函数声明仅列出一种。提供两个接口是为了方便能同时使用两种不同的字符集。
常用的字体名称宋体，楷体 _GB2312，隶书，黑体，幼圆，新宋体，仿宋 _GB2312，Fixedsys，Arial，Times New Roman
具体可用名查阅你系统已安装字体。

nEscapement
字符串的书写角度，单位 0.1 度。

nOrientation
每个字符的书写角度，单位 0.1 度。

nWeight
字符的笔画粗细，范围 0~1000。0 表示默认粗细。使用数字或下表中定义的宏均可：

	粗细值
FW_DONTCARE	0
FW_THIN	100
FW_EXTRALIGHT	200
FW_ULTRALIGHT	200
FW_LIGHT	300
FW_NORMAL	400

FW_REGULAR	400
FW_MEDIUM	500
FW_SEMIBOLD	600
FW_DEMIBOLD	600
FW_BOLD	700
FW_EXTRLD	800
FW_ULTRABOLD	800
FW_HEAVY	900
FW_BLACK	900

bItalic
是否斜体， true / false 。

bUnderline
是否有下划线， true / false 。

bStrikeOut
是否有删除线， true / false 。

fbCharSet
指定字符见 LOGFONT 结构体)。

fbOutPrecision
指定文字的输出精度 (详见 LOGFONT 结构体)。

fbClipPrecision
指定文字的剪辑精度 (详见 LOGFONT 结构体)。

fbQuality
指定文字的输出质量 (详见 LOGFONT 结构体)。

fbPihAndFamily
指定以常规方式描述字体的字体系列 (详见 LOGFONT 结构体)。

font
指向 LOGFONT 结构体的指针。

返回值：
(无)

示例：

```
// 设置当前字体为高 16 像素的“宋（忽略行距）”。
setfont ( 6, 0, "宋体" );
outtextxy ( 0, 0, "测试" );
/ 设置输出效果为抗锯齿 ( LOGFONT 是 MBCS 版本， LOGFONT 是 UTF16 版本 )
LOGFONT f;
getfont ( &f ); // 获取当前字体设置
f.lfHeight = 48; // 设置字体高度为 48 ( 包含行距 )
```

```
strcpy (f.lfFaceName, "黑体 ");           // 设置字体为 “ 黑体 ”
f.lfQuality = ANTIALIAS_QUALITY;         // 设置输出效果为抗锯齿
setfont (&f);                             // 设置字体样式
outtextxy (0,50, " 抗锯齿效果 ");
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.8 setttextjustify

```
vd setttextjustify
    int  horiz,
    int  vert,
    PIMAGEpimg = NULL
);
```

参数：

horiz

横向对齐方式，可选值 LEFT_TEXT (默认), CENTER_TEXT, RIGHT_TEXT

vert

向对齐方式，可选值 TOP_TEXT (默认), CENTER_TEXT, BOTTOM_TEXT

textstring

要输出的字符串的指针

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.9 textheight

参数：

textsng

指定的字符串指针。

返回值：

该字符串实际占用的像素高度。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.10 textwidth

```
int textwidth (
    LPCTSTRtextstring,
    PIMAGEpimg = NULL
);
```

参数：

textstring

指定的字符串指针。

返回值：

该字符串实际占用的像素宽度。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.11 xyprintf

```
void xyprintf (
```

```
    int x,
    int y,
    int w,
    int h,
    LPCSTRtextstring,
    ...
);
void xyprintf (
    int x,
    int y,
    int w,
    int h,
    LPCWSTRtextstring,
    ...
);
```

参数：

x, y, w,
要输出字符串所在的矩形区域

textstring
要输出的字符串的指针。

...
要输出的内容的参数，类似 printf 。

返回值：

（无）

示例：

无

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.5.12 xyprintf

```
void xyprintf (
    int x,
    int y,
    LTR textstring,
```



```
...
);
void xyprintf (
    int x,
    int y,
    LPCWSTR textstring,
    ...
);
```

参数：

x
字符串输出时头字母的 x 轴的坐标值

y
字符串输出时头字母的 y 轴的坐标值。

textstring
要输出的的指针。

...
要输出的内容的参数，类似 printf 。

返回值：

（无）

示例：

无

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6 图像处理相关函数

5.6.1 图像处理相关函数

函数或数据	说明
getimage (See 5.6.2)	从屏幕 文件 / 资源 / IMAGE 对象中获取图像
PIMAGE 对象 (See 5.6.4)	保存图像的对象
imagefilter_blurring (See 5.6.3)	对指定图像进行图像模糊滤镜操作
putimage (See 5.6.5)	在屏幕上绘制指定图像

putimage_alphablend (See 5.6.6)	在屏幕上以半透明方式绘制指定图像
putimage_transparent (See 5.6.8)	在屏幕上以透明方式绘制指定图像
putie_alphatransparent (See 5.6.7)	在屏幕上以透明 / 半透明方式绘制指定图像
三元光栅 (See 5.6.9)	介绍三元光栅操作码

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.2 getimage

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [图像处理相关函数](#) (See 5.6.1)- > [getimage](#)

功能：

这个函数的四个重载分别用于从屏幕 / 文件 / 源 / IMAGE 对象中获取图像

声明：

```
// 屏幕获取图像
void getimage (
    PIMAGEpDstImg,      // 保存图像的 IMAGE 对象指针
    int srcX,            // 获取图像的区域左上角 x 坐标
    int srcY,            // 要获取图像的区域左上角 y 坐标
    int srcWidth,        // 要获取图像的区域宽度
    int srcHeight        // 要获取图像的区域高度
);

// 从图片文件获取图像 (png/bmp/jpg/gif/emf/wmf/ico)
void getimage (
    PIMAGEpDstImg,      // 保存 IMAGE 对象指针
    LPCTSTRpImgFile,    // 图片文件名
    int zoomWidth = 0,   // 设定图像缩放至的宽度 ( 0 表示默认宽度，不缩放 )
    int zoomHeight = 0   // 设定图像缩放至的高度 ( 0 表示默认高度，不缩放 )
);

// 从资源文件获取图像 (png/bmp/jpg/gif/emf/wmf/ico)
void getimage (
    PIMAGEpDstImg,      // 保存图像的 IMAGE 对象指针
    LPCTSTRpResType,    // 资源类型
```

```

    LPCTSTR ResName,      // 资源名称
    int zoomWidth = 0,     // 设定图像缩放至的宽度 ( 0 表示默认宽度, 不 )
    int zoomHei = 0       // 设定图像缩放至的高度 ( 0 表示默认高度, 不缩放 )
);
// 从另一个 IMAGE 对象中获取图像
void getimage (
    PIMAGEpDstImg,        // 保存图像的 IMAGE 对象指针
    const IMAGE *pSrcImg,  // 源图像 IMAGE 对象
    int srcX,              // 要获取图像的区域左上角 x 坐标
    int srcY,              // 要获取图像的区域左上角 y 坐标
    int srcWidth,          // 图像的区域宽度
    int sHeight            // 要获取图像的区域高度
);

```

参数：

mg

(详见各重载函数原型内的注释)

返回值：

(无)

示例：

请参考 putie 函数示例。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.3 imagefilter blurring

```

int imagefilter_blurring (
    PIMAGEimgdest,
    int intensity,
    int alpha,
    int nXOriginDest = 0,
    int nYOriginDest = 0,
    int nWidthDest = 0,
    int nHeightDest = 0
);

```

参数：

imgdes

要进行模糊操作的图片，如果为 NULL则表示操作窗口上的图片

intensity

模糊度，值越大越模糊。当值在 0x0 - 0x7F 之间时四向模糊；当值在 0x80 - 0xFF 之间时，为八向模糊，运算量会大一倍

alpha

图像亮度。取值为 0x100 表示亮度不变，取值为 0x0 表示图像变成纯黑

nXOriginDest, nYOriginDest, nWidthDest, nHeightDest

描述要进行此操作的矩形区域。如果 nWidthDest 和 nHeightDest 为 0，表示操作整张图片。

返回值：

成功返回 0，否则返回非 0，若 gdest 传入错误，会引发运行时异常。

示例：

（无）。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.4 PIMAGE

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [图像处理相关函数 \(See](#)

5.6.1)- > PIMAGE

功能：

保存图像的对象

示例：

以下语句可以创建一个名为 pimg 的 PIMAGE 对象，尺寸为 100x50：

```
PIMAGE pimg = newimage(100, 50);
```

注意，不需要时要 delimage(pimg); 来释放掉，切记。

更多示例请参考 `putimage` 函数示例。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.5 putimage

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [图像处理相关函数 \(See 5.6.1\)](#)- > `putimage`

功能：

函数的几个重载用于在屏幕或另一个图像上绘制指定图像。

声明：

// 绘制图像到屏幕

```
void putimage (
    int dstX,           // 绘制位置的 x 坐标
    int dstY,           // 绘制位置的 y 坐标
    PIMAGEpSrcImg,      // 要绘制的 IMAGE 对象指针
    DWORD dwRop = SRCCOPY // 三元光码（详见备注）
);
```

// 绘制图像到屏幕（指定

```
void putimage (
    int dstX,           // 绘制位置的 x 坐标
    int dstY,           // 绘制位置的 y 坐标
    int dstWidth,       // 绘制的宽度
    int dstHeight,      // 绘制的高度
    PIMAGEpSrcImg,      // 要绘制的 IMAGE 对象指针
    int srcX,           // 绘制内容在 IMAGE 对象中的左上角 x 坐标
    int srcY,           // 绘制内容 IMAGE 对象中的左上角 y 坐标
    DWORD dwRop = SRCCOPY // 三元光栅操作码（详见备注）
);
```

// 绘制图像到屏幕（拉伸）

```
void putimage (
    int dstX,           // 绘制位置的 x 坐标
    int dstY,           // 绘制位置的 y 坐标
    int dstWidth,       // 绘制的宽度
    int dstHeight,      // 绘制的高度
    PIMAGEpSrcImg,      // 要绘制的 IMAGE 对象指针
    int srcX,           // 绘制内 IMAGE 对象中的左上角 x 坐标
```

```

        i srcY,                // 绘制内容 IMAGE 对象中的左上角 y 坐标
        int srcWidth,          // 绘制内容在源 IMAGE 对象中的宽度
        int srcHeight,         // 绘制内容在源 IMAGE 对象中的高度
        DWORD dwRop = SRCCOPY // 三元光栅操作码 (详见备注)
    );
    绘制图像到另一图像
void putimage (
    PIMAGEpDstImg,            // 标 IMAGE 对象指针
    int dstX,                  // 绘制位置的 x 坐标
    int dstY,                  // 绘制位置的 y 坐标
    PIMAGEpSrcImg,            // 源 IMAGE 对象指针
    DWORD dwRop = SRCCOPY // 三元光栅操作码 (详见备注)
);
// 绘制图像到另一图像 (指定宽高)
void putimage (
    PIMAGEpDstImg            // 目标 IMAGE 对象指针
    int dstX,                  // 绘制位置的 x 坐标
    int dstY,                  // 绘制位置的 y 坐标
    int dstWidth,              // 绘制的宽度
    int dstHeight,             // 绘制的高度
    PIMAGEpSrcImg,            // 源 IMAGE 对象指针
    int srcX,                  // 绘制内容在源 IMAGE 对象中的左上角 x 坐标
    int rcY,                   // 绘制内容在源 IMAGE 对象中的左上角 坐标
    DWORD dwR = SRCCOPY // 三元光栅操作码 (详见备注)
);
// 绘制图像到另一图像 (拉伸)
void putimage (
    PIMAGEpDstImg,            // 目标 IMAGE 对象指针
    t dstX,                    // 绘制位置的 坐标
    int dstY,                  // 绘制位置的 y 坐标
    int dstWidth,              // 绘制的宽度
    int dstHeight,             // 绘制的高度
    PIMAGEpSrg,               // 源 IMAGE 对象指
    int srcX,                  // 绘制内容在源 IGE 对象中的左上角 x 坐标
    int srcY,                  // 绘制内容在源 IMAGE 对象中的左上角 y 坐标
    int srcWidth,              // 绘制内容在源 IMAGE 对象中的宽度
    int srcHeight,             // 绘制内容在源 IMAGE 对象中的高度
    DWORD dwRop = SRCCOPY // 光栅操作码 (详见备注)
);

```

参数：

(详见各重载函数原型内的注释)

备注：
三元光栅操作码（即位操作模式），支持的 256 种三元光栅操作码，常用的几种如下：

值	含义
DSTINVERT	绘制出的像素颜色 = NOT 颜色
MERGECOPY	绘制素颜色 = 图像颜色 AND 当前填充颜色
MERGEPAINT	绘制出的像素颜色 = 屏幕颜色 OR (NOT 图像颜色)
NOTSRCCOPY	绘制出的像素颜色 = NOT 图像颜色
NOTSRCERASE	绘制出的像素颜色 NOT (屏幕颜色 OR 图像颜色)
PATCOPY	绘制出的像素颜色 = 当前填充颜色
PATINVERT	绘制出的像素颜色 = 屏幕颜色 XOR 当前填充颜色
PATPAINT	绘制出的像素颜色 = 屏色 OR ((NOT 图像颜色) OR 当前填充颜色)
SRCAND	绘制出的像素颜色 = 屏幕颜色 AND 图像颜色
SRCCOPY	绘制出的像素颜色 = 图像颜色
CERASE	绘制出的像素颜色 = (NOT 屏幕颜色) AN 图像颜色
SRCINVERT	绘制出的像素颜色 = 屏幕颜色 XOR 图像颜色
SRCPAINT	绘制出的像素颜色 = 屏幕颜色 OR 图像颜色

注：1. AND / OR / NOT / XOR 为布尔位运算。 2. " 屏幕颜色 " 指绘制所经过的屏幕像素点的颜色。 3. " 图像颜色 " 是指通过 IMAGE 对象中的图像的颜色。 4. " 当前填充颜色 " 是指通过 setfillstyle 设置的用于当前填充的颜色。 [查看全部的三元光栅操作码](#)请详见：[三元光栅操作码](#)。

返回值：
(无)

示例：

```
以下局部取 c:\test.jpg 绘制在屏幕左上角：
PIMAGEimg = newimage();
getimage (img, "c:\\tespg" );
putimage (0, , img);
delimage (img);
以下局将屏幕 (0,0) 起始的长宽各 100 像素的图像拷贝至 (200,200) 位置：
PIMAGEimg = newimage();
getimage (img, 0, 0, 100, 100);
putimage (200, 200, img);
```

delimage (img);

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.6 putimage alphablend

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [图像处理相关函数](#) (See 5.6.1)- > pmage_alphablend

功能：

这个函数用于对两张图片进行半透明混合混合结果写入目标图片。

声明：

```
int putimage_alphablend (
    PIMAGEimgdest,           // handle to dest
    PIMAGEimgsrc,            // handle to source
    int nXOriginDest,         // x-coord of destination upper-left corner
    int nYOriginDest,         // y-coord of destination upper-left corner
    unsigned char alpha,      // alpha
    int nXOriginSrc = 0,      // x-crd of source upper-left corner
    int nYOriginSrc = 0,      // y-coord of source upper-left corner
    int nWidthSrc = 0,        // width of source rectangle
    int nHeightSrc = 0        // height of source rectangle
);
```

参数：

imgdest
要进行半透明混合的目标图片，如果为 NUL 示操作窗口上的图片

imgsrc
要进行半透明混合的源图片，该操作不会改变源图片

OriginDest, nYOriginDest
要开始进行混合的目标图片坐标，该坐标是混合区域的左上角

alpha

透明度值，如果为 0x0，表示源图片完全透明，如果为 0xFF，表示源图片完全不透明。

OriginDest, nYOriginDest, nWidthDest, nHeightDest

描述要进行此操作的源图矩形区域。如果 nWidthDest 和 nHeightDest 为 0，表示操作整张图片。

返回值：

成功返回 0，否则返回非 0，若 imgdest 或 imgsrc 传入错误，会引发运行时异常。

示例：

(无)。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.7 putimage alphastransparent

[主页 \(See 1.\)](#) - > [库函数目录 \(See 5.1\)](#) - > [图像处理相关函数 \(See 5.6.1\)](#) - > putimagealphatransparent

功能：

这个函数用于对两张图片进行透明 / 半透，并把混合结果写入目标图片。

声明：

```
int putimage_alphastransparent (
    PIMAGE imgdest,           // handle to dest
    IMAGE imgsrc,             // handle source
    int nXOriginDest,         // x-coord of destination upper-left corner
    int nYOriginDest,         // y-coord of destination upper-l corner
    COLORREF transparent,     // color to make transparent
    unsigned char alpha,      // alpha
    int nXOriginSrc = 0,      // x-coord of source upper-left corner
    int nYOriginSrc = 0,      // y-coo of source upper-left corner
    int nWidthSrc = 0,        // width of soce rectangle
    int nHeightSrc = 0        // height of source rectangle
);
```

参数：

imgdest

要进行半透明混合的目标图片，如果为 NULL则表示操作窗口上的图片

imgsrc

要进行半透明混合的源图片，该操作不会改变源图片

nXOriginDest, nYOriginDest

要开始进行混合的目标图片坐标，该坐标是混合区域的左上角

crTransparent

关键色。源图片上为该颜色值的像素，将忽略，不会改写目标图片上相应位置的像素。

alpha

透明度值，如果为 0x0，表示源图片完全透明，如果为 0xFF，表示源图片完全不透明。

nXOriginDest, nYOriginDest, nWidthDest, nHeightDest

描述要进行的源图矩形区域。如果 nWidthDest 和 nHeightDest 为 0，表示操作整张图片。

返回值：

成功返回 0，否则返回非 0，若 imgt 或 imgsrc 传入错误，会引发运行时异常。

示例：

（无）。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.8 putimage transparent

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [图像处理相关函数](#) (See

5.6.1)- > putimage_transparent

功能：

这个函数用于图片进行透明混合，并把混合结果写入目标图片。

声明：

```
t putimage_transparent (
    PIMAGEimgdest,           // handle to dest
    PIMAGEimgsrc,            // handle to sour
    int  nXOriginDest,        // x-coord of destination upper-left corne
    int  nYOriginDest         // y-coord of destination upper-left corne
    COLORREF crTransparent,   // color to make transparent
    int  nXOriginSrc = 0,     // x-coord of source upper-left corner
    int  nYOriginSrc = 0,     // y-coord of source upper-left corner
    int  nWidthSrc = 0,       // width of source rectangle
    int  nHeightSrc = 0       // height of source rectangle
);
```

参数：

imgdest

要进行透明混合的目标图片，如果为 NULL则表示操作窗口上的图片

imgsrc

要进行透明混合的源图片，该操作不会改变源图片

nXOriginDest, nYOriginDest

要开始进行混合的目标图片坐标，该坐标是混合区域的左上角

crTransparent

关键色。源图片上为该颜色值的像素，将忽略，不会改写目标图片上相应位置的像素。

nXOriginDest, OriginDest, nWidthDest, nHeightDest

描述要进行此操作的源图矩形区域。如果 nWidthDest 和 nHeightDest 为示操作整张图片。

值：

成功返回 0，否则返回非 0，若 imgdest 或 imgsrc 传入错误，会引发运行时异常。

示例：

(无)。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.6.9 三元光栅操作码

[主页 \(See 1.\)](#)-
 >
 [库函数目录 \(See 5.1\)](#)-
 >
 [图像处理相关函数 \(See 5.6.1\)](#)-
 >
 三元光栅操作码

这篇补充文档列 putimage 函数支持的所有三元光栅操作码。
三元光栅操作码定义了源图像与屏幕图像的位合并形式，这个合并形式是以下三个操作数对应像素的布尔运算：

操作数	含义
D	屏幕图像
P	当前填充颜色
S	源图像

布尔运算符包括以下几种：

操作	含义
a	位的 AND 运算 (双目运算)
n	位的 NOT 运算 (单目运算)
o	位的 OR 运算 (双目运算)
x	位的 XOR 运算 (双目运算)

的布尔操作都采用逆波兰表示法， 例如，“当前填充颜色 or 源图像”可表示为：PS_o（当然 SP_o 也是等价的，这里只列举出了其中一种等价格式）
元光栅操作码是 32 位 int 类型，其高位字是布尔操作索引，低位字是操作码。
布尔操作索引的 16 个位中，高 8 位用 0 填充，低 8 位是当前色、源图像和屏幕的布尔操作结果。例如， PS_o 和 DPS_{oo} 的操作索引如下：

P	S	D	PS _o	DPS _{oo}
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0		1

1	1	1	1	1
操作索引：			00FCh	00FEh

上例中， PSo 的操作索引是 00FC（从下往上读），DPS 的是 00FE。这些值定义了相应的三元光栅操作码在“三元光栅操作码”表格中的位置， PSo 在 252（00FCh）行，DPSoo 在 254（0FEh）行。常用的三元光栅操作码已经定义了常量名，程序中可以直接使用。

三元光栅操作码

布尔功能（16 进制）	光栅操作（16 进制）	布尔功能的逆波兰	常量名
00	00000042	0	BLACKNESS
01	00010289	DPSoon	
02	00020C89	DPSona	
03	0003A	PSon	
04	00040C88	SDPona	
05	000500A9	DPon	
06	00060865	PDSxnon	
07	000702C5	PDSaon	
08	00080F08	Pnaa	
09	00090245	PDSxon	
0A	000A0329	DPna	
0B	000B0B2A	PSDnaon	
0C	000C0	SPna	
0D	000D0B25	PDSnaon	
0E	000E08A5	PDSonon	
F	000F0001	Pn	
10	00100C85	PDSona	
11	001100A6	DSon	NOTSRCER
12	00120868	SDPxnon	
13	001302C8	SDPaon	
14	00140869	DPSxnon	
15	001502C9	DPSaon	
16	00165CCA	PSDnaxx	
17	00171D54	SSPxDxn	
18	00180D59	SPxPDxa	
19	00191CC8	SDPSanaxn	

1A	001A06C5	PDSPaox	
1B	001B0768	SDPSxaxn	
1C	001CA	PSDPaox	
1D	001D0766	DSPDxaxn	
1E	001E01A5	PDSox	
1F	001F0385	PDSoan	
20	00200F09	DPSnaa	
21	00210248	SDPxon	
22	00220326	DSna	
23	00230B24	SPDnaon	
24	00240D55	SPxDSxa	
25	00251CC5	PDSPanaxn	
26	0026C8	SDPSaox	
27	00271868	SDPSxnox	
28	00280369	DPSxa	
29	002916CA	PSDPSaoxxn	
2A	002A0CC9	DPSana	
2B	002B1D58	SSPxPDxaxn	
2C	002C0784	SPDSoax	
2D	002D060A	PSDnox	
E	002E064A	PSDPxox	
2F	002F0E2A	PSDnoan	
30	0030032A	PSna	
31	00310B28	SDPnaon	
32	00320688	SDPSoox	
33	30008	Sn	NOTSRCCOPY
34	406C4	SPDSaox	
35	00351864	SPDSxnox	
36	003601A8	SDPox	
37	00370388	SDPoan	
38	0038078A	PSDPoax	
39	00390604	SPDnox	
3A	003A0644	SPDSxox	
3B	003B0E24	SPDnoan	

3C	003C004A	PSx	
3D	003D18A4	SPDSonox	
3E	003E1B24	SPDSnaox	
3F	003F00EA	PSan	
40	00400F0A	PSDnaa	
41	00410249	DPSxon	
42	00420D5D	SDxPDxa	
43	00431CC4	SPDSanaxn	
44	00440328	SDna	SRCERASE
45	00450B29	DPSnaon	
46	004606C6	DSPDaox	
47	0076A	PSDPxaxn	
48	00480368	SDPxa	
49	004916C5	PDSPDaoxxn	
4A	004A0789	DPSPDoax	
4B	004B0605	PDSnox	
4C	004C0CC8	SDPana	
4D	004D1954	SDSxoxn	
4E	004E0645	PDSPxox	
4F	004F0E25	PDSnoan	
50	00500325	PDna	
51	00510B26	DSPnaon	
52	005C9	DPSPDaox	
53	00530764	SPDSxaxn	
54	005408A9	DPSonon	
55	00550009	Dn	DSTINVERT
56	005601A9	DPSox	
57	00570389	DPSoan	
58	0058078	PDSPoax	
59	00590609	DPSnox	
5A	005A0049	DPx	PATINVERT
5B	005B18A9	DPSPDonox	
5C	005C0649	DPSPDxox	
5D	005D0E29	DPSnoan	

5E	005E1B29	DPSDnaox	
5F	005F00E9	DPan	
60	00600365	PDSxa	
61	006116C6	DSPDSaoxxn	
62	00620786	DSax	
63	00630608	SDPnox	
64	00640788	SDPSoax	
65	00650606	DSPnox	
66	00660046	DSx	SRCINVERT
67	006718A8	SDPSonox	
68	006858A6	DSPDSonoxxn	
69	90145	PDSxxn	
6A	006A01E	DPSax	
6B	006B178A	PSDPSoaxxn	
6C	006C01E8	SDPax	
6D	006D1785	PDSPDoaxxn	
6E	006E1E28	SDPSnoax	
6F	006F0C	PDSxnan	
70	00700CC5	PDSana	
71	00711D5C	SSDxPDxaxn	
72	00720648	SDPSxox	
73	00730E28	SDPnoan	
	00740646	DSPDxox	
75	00750E26	DSPnoan	
76	00761B28	SDPSnaox	
77	007700E6	DSan	
78	007801E5	PDSax	
79	0791786	DSPDSoaxxn	
7A	007A1E29	DPSDnoax	
7B	007B08	SDPxnan	
7C	007C1E24	SPDSnoax	
7D	007D0C69	DPSxna	
7E	007E0955	SPxDSxo	
7F	007F03C9	DPSaan	

80	00809	DPSaa	
81	00810975	SPxDSxon	
82	00820C49	DPSxna	
83	00831E04	SPDSnoaxn	
	00840C48	SDPxna	
85	00851E05	PDSPnoaxn	
86	008617A6	DSPDSoaxx	
87	008701C5	PDSaxn	
88	008800C6	DSa	SRCAND
	00891B08	SDPSnaoxn	
8A	008A0E06	DSPnoa	
8B	008B0666	DSPDxoxn	
8C	008C0E08	SDPno	
8D	D0668	SDPSxoxn	
8E	008E1D7C	SSDxPDxax	
8F	008F0CE5	PDSanan	
90	00900C45	PDSxna	
91	00911E08	SDPSnoaxn	
92	009217A9	DPSPDpoaxx	
93	009301C4	SPDaxn	
94	009417AA	PSDPSoaxx	
95	009501C9	DPSaxn	
96	00960169	DPSxx	
97	0097588A	PSDPSonoxx	
98	00981888	SDPSonoxn	
99	00990066	DSxn	
9A	009A0709	DPSnax	
9B	009A8	SDPSoaxn	
9C	009C0704	SPDnax	
9D	009D07A6	DSPDoaxn	
9E	009E16E6	DSPDSoaxx	
9F	009F0345	PDSxan	
A0	00A000C9	DPa	
A1	00A11B05	PDSPnaoxn	

A2	00A20E09	DPSnoa	
A3	00A3066	DPsDxoxn	
A4	00A41885	PDSPonoxn	
A5	00A50065	PDxn	
A6	00A60706	DSPnax	
A7	00A707A5	SPoaxn	
A8	00A803A9	DPSoa	
A9	00A90189	DPSoxn	
	00AA0029	D	
AB	00AB0889	DPSono	
AC	00AC0744	SPDSxax	
	00AD06E9	DPsDaoxn	
AE	00AE0B06	DSPnao	
AF	00AF0229	DPno	
B0	00B00E05	PDSnoa	
B1	00B10665	PDSPxoxn	
B2	00B21974	SSPxDSxox	
B3	00B30CE8	SDPanax	
B4	00B4070A	nax	
B5	00B507A9	DPsDdoxn	
B6	00B616E9	DPsDPaoxx	
B7	00B70348	SDPxan	
B8	00B8074A	PSDP	
B9	00B906E6	DSPDaoxn	
BA	00BA0B09	DPSnao	
BB	00BB0226	DSno	MERGEPAINT
BC	C1CE4	SPDSanax	
BD	00BD0D7D	SDxPDxan	
BE	00BE0269	DPSxo	
BF	00BF08C9	DPSano	
C0	00C000CA	PSa	GECOPY
C1	00C11B04	SPDSnaoxn	
C2	0C21884	SPDSonoxn	
C3	00C3006A	PSxn	

C4	00C40E04	SPDnoa	
C5	00C50664	SPDSxoxn	
C6	00C60708	SDPnax	
C7	00C707AA	PSDPoaxn	
C8	00C803A8	SDPoa	
C9	00C90184	SPDoxn	
CA	00CA0749	DPSDxax	
CB	00CB06E4	SPDSaoxn	
CC	00CC0020	S	SRCCOPY
CD	00CD0888	SDPono	
CE	00CE0B08	SDPnao	
CF	00CF0224	SPno	
D0	00D00E0A	PSDnoa	
D1	00D1066A	PSDPxoxn	
D2	00D205	PDSnax	
D3	00D307A4	SPDSsoaxn	
D4	00D41D78	SSPxPDxax	
5	00D50CE9	DPSanan	
D6	00D616EA	PSDPSaox	
D7	D70349	DPSxan	
D8	00D80745	PDSPxax	
D9	00D906E8	SDPSaoxn	
DA	00DA1CE9	DPSDanax	
DB	00DB0D75	SPxDSxan	
DC	00DC0B04	SPDnao	
DD	00DD0228	SDno	
DE	00DE0268	SDPxo	
DF	00DF08C8	SDPano	
E0	00E003A5	PDSoa	
	00E10185	PDSoxn	
E2	00E20746	DSPDxax	
E3	00E306	PSDPaoxn	
E4	00E40748	SDPSxax	
E5	00E506E5	PDSPaoxn	

E6	00E61CE8	SDPSanax	
E7	00E70D79	SPxPDxan	
E8	00E81D74	SSPxDSxax	
E9	00E95CE6	DSPDSanaxxn	
EA	00EA02E9	DPSao	
EB	00EB0849	DPSxno	
EC	00EC02E8	SDPao	
ED	00ED0848	SDPxno	
EE	00EE0086	DSO	SRCPAINT
EF	00EF0A08	SDPnoo	
F0	00F00021	P	PATCOPY
F1	10885	PDSono	
F2	00F20B05	PDSnao	
F3	00F3022A	Sno	
F4	00F40B0A	PSDnao	
F5	00F50225	PDno	
F6	00F60265	PDSxo	
F7	00F708C5	PDSano	
F8	00F802E5	PDSao	
F9	00F90845	PDSxno	
FA	00FA0089	DPo	
FB	00FB0A09	DPSnoo	PATPAINT
FC	00FC008A	PSO	
FD	FD0A0A	PSDnoo	
FE	00FE02A9	DPSoo	
FF	00FF0062	1	WHITSS

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7 键盘鼠标输入函数

5.7.1 键盘鼠标输入函数

函数或数据	说明
FlushMouseMsgBuffer (See 5.7.2)	清空鼠标消息缓冲区（数）
getch (See 5.7.3)	获取一个键盘字符输入。 如果当前没有输入的字符就一直等待
getkey (See 5.7.4)	获取一个键盘消息。如果当前没有键盘消息就一直等待
getmouse (See 5.7.5)	获取一个鼠标消息。如果当前鼠标消息队列中没有，就一直等待
GetMouseMsg (See 5.7.6)	（过）获取一个鼠标消息。如果当前鼠标消息队列中没有，就一直等待
kbhit (See 5.7.7)	检测当前是否有键盘字符输入，一般与 getch 搭配使用
kbmsg (See 5.7.8)	检测当前是否有键盘消息，一般与 getkey 搭配使用
key_msg (See 5.7.9)	保存键盘消息的结构体
keystate (See 5.7.10)	判断某按键是否按下
mousemsg (See 5.7.12)	检测当前是否有鼠标消息
mouse_g 结构体 (See 5.7.11)	保存鼠标消息的结构体
MOUSEMSG 结构体 (See 5.7.13)	（不建议使用）保存鼠标消息的结构体
mousepos (See 5.7.14)	获取当前鼠标位置
showmouse (See 5.7.15)	设置鼠标显示

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.2 FlushMouseMsgBuffer

```
void FlushMoeMsgBuffer ();
```

参数：
)

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.3 getch

参数：

(无)

返回值：

如果存在键盘字符输入，返回按键键码；否则不返回一直等待。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.4 getkey

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [键盘鼠标输入函数 \(See](#)

5.7.1)- > getkey

功能：

这个函数用于获取键盘消息，如果当前没有消息，则等待。

声明：

```
key_msg getkey ();
```

参数：

(无)

返回值：

返回 key_msg 结构体

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.5 getmouse

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [键盘鼠标输入函数 \(See 5.7.1\)](#)- > getmouse

功能：

这个函数用于获取一个鼠标消息。如果当前鼠标消息队列中没有，就一直等待。

声明：

```
mouse_msg getmouse ();
```

参数：

(无)

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by

TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.6 GetMessage

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [键盘鼠标输入函数 \(See 5.7.1\)](#)- > GetMessage

功能：

这个函数用于获取一个鼠标消息。如果当前鼠标消息队列中没有，就一直等待。

声明：

```
MOUSEMSG GetMessage();
```

参数：

(无)

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.7 kbhit

[主页 \(See 1.\)](#)- > [库函数目录 \(See 5.1\)](#)- > [键盘鼠标输入函数 \(See 5.7.1\)](#)- > kbhit

功能：

这个函数用于检测当前是否有键盘字符输入。

声明：

int kbhit ();

参数：

(无)

返回值：

如果存在键盘字符输入，返回 1；否则返回 0。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.8 kbmsg

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [键盘鼠标输入函数](#) (See 5.7.1)- > kbmsg

功能：

这个函数用于检测当前是否有键盘消息。

声明：

int kbmsg();

参数：

(无)

返回值：

如果存在键盘消息，返回 1；否则返回 0。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.9 key_msg 结构体

```
typedef struct    key_msg {
    UIN msg;
    UINT y;
    UINT flags;
}key_msg;
```

成员：

msg
指定鼠标消息类型，可为以下值：

值	含义
key_msg_down	键盘按下消息。
key_msg_up	键盘弹起消息。
key_msg_char	键盘字符输入消息。

key
如果是按下和弹起的消息，则表示按键虚拟键码，否则为 gbk 编码字符消息

flags
按键参数，可能值的组合：

值	含义
key_flag_shift	同时按下了 shift
key_flag_rl	同时按下了 control

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.10 keystate

```
int keystate (int key);
```

参数：

key

虚拟键码，一个 Windows 定义的，能与键盘按键一一对应的码表。如果是字母键或者数字键，则其虚拟键码与 ASCII 字符值相同，比如 'A' 键，它的虚拟键码也是 'A'，但不能是 'a'。小键盘上的数字则用类似 VK_NUMPAD0 的宏表示，详细可以查看 VK_XXX 这一系列宏的定义。

返回值：

返回非 0 表示这个按键按下了，返回 0 表示没有按下。全局有效，即使窗口没有得到输入焦点，一样照样取得键盘的实际状态。

示例：

```
if (keystate (VK_ESCAPE))
{
    // ESC 键按下了
}
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.11 mouse_msg 结构体

```
typedef struct mouse_msg {
    UINT msg;
    INT x;
    INT y;
    UINT flags;
    INT wheel;
}mouse_msg;
```

成员：

x

当前鼠标 x 坐标

y

当前鼠标 y 坐标

is_move()

是否鼠标移动消息，类型为 `bool`

`is_down()`

是否鼠标按键按下消息，类型为 `bool`

`is_up()`

是否鼠标按键放开消息，类型为 `bool`

`is_left()`

是否鼠标左键消息，类型为 `bool`

`is_mid()`

是否鼠标中键消息，类型为 `bool`

`is_right()`

是否鼠标右键消息，类型为 `bool`

`is_wheel()`

是否鼠标滚轮滚动消息，类型为 `bool`

`wheel`

鼠标滚轮滚动值，一般情况下为 `120` 的倍数或者约数。

`msg`

不建议直接访问

`flags`

不建议直接访问

示例：

参见入门基础示例 `15`

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.12 mousemsg

`int mousemsg();`

参数：

(无)

返回值：

如果存在鼠标消息，返回 `1`；否则返回 `0`。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.13 MOUSEMSG 结构体

```
struct    MOUSEMSG
{
    UINT  uMsg;           // 当消息
    int   mkCtr           // Ctrl   键是否按下
    int   mkShift;        // Shift   键是否按下
    int   mkLButton;      // 鼠标左键是否按下
    int   mkMButton;      // 鼠标中键是否按下
    int   mkRButton;      // 鼠标右键是否按下
    int   x;              // 当前鼠标 x 坐标
    int   y;              // 当前鼠标 y 坐标
    int   wheel;          // 鼠标滚轮滚动值
};
```

成员：

uMsg：
指定鼠标消息类型，可为以下值：

值	含义
WM_MOUSEMOVE	鼠标移动消息。
WM_MOUSEWHEEL	鼠标滚轮拨动消息。
WM_LBUTTONDOWN	左键按下消息。
WM_LBUTTONUP	左键弹起消息。
WM_LBUTTONDBLCL	左键双击消息。
WM_MBUTTONDOWN	中键按下消息。
WM_MBUTTONUP	中键弹起消息。
WM_MBUTTONDBLC	中键双击消息。
WM_RBUTTONDOWN	右键按下消息。
WM_RBUTTONUP	右键弹起消息。
WM_RBUTTONDBLCLK	右键双击消息。

mkCtrl

Ctrl 键是否按下

mkShift

Shift 键是否按下

mkLButton

鼠标左键是否按

mkMButton

鼠标中键是否按下

mkRButton

鼠标右键是否按下

x

当前鼠标 x 坐标

y

当前鼠标 y 坐标

heel

鼠标滚轮滚动值，一般情况下为 120 的倍数或者约数。

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.14 mousepos

```
int mousepos(int x, int *y);
```

参数

x

用来接收横坐标

y

用来接收纵坐标

返回值：

0

示例：

（无）

Caution: Some letters were intendedly removed from the document because It was created by

TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.7.15 showmouse

```
int showmouse(int bShow);
```

参数：

bShow
为 0 则不显示，非 0 为显示。默认显示。

返回值：

返回调用时设置的值，第一次调用的话返回 1。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.8 数学函数

5.8.1 数学函数

函数或数据	说明
rote_point3d_x	一个 3d 点绕 x 轴旋转
rotate_point3d_y	把一个 3d 点绕 y 轴旋转
rotate_point3d_z	把一个 3d 点绕 z 轴旋转
VECTOR3D	的 3d 向量类，以下为类的成员简介
CTOR3D::operator =	向量复制
VECTOR3D::operator +	3d 向量加法
VECTOR3D::operator -	3d 向量减法
VEC3D::operator *	与浮点数相乘时为向量缩放，与向量相乘时为点乘
VECTOR3D::operator &	向量叉乘
VECTOR3D::GetAngel	计算两个 3d 向量的夹角
ECTOR3D::GetModule	计算 3d 向量的模

VECTOR3D::GetSqrModule	计算 3d 向量模的平方
VECTOR3D::Rotate	3d 向量绕另一任意向量旋转，或者按指定旋转角旋转
VECTOR3D::SetModule	在保持方向不变的情况下把 3d 向量长度改为设定值

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.9 其它函数

5.9.1 其它函数

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > 其它

目录：

函数或数据	说明
getfps (See 5.9.2)	获取当前窗口刷新率 (FPS = Frame Per Second)
GetHWnd (See 5.9.3)	获取当前窗口句柄
inputbox_getline (See 5.9.4)	使用对话框让用户输入一个字符串

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.9.2 getfps

```
float  getfps (
    t  flag = 1
);
```

参数：

flag
仅能为 0 或者 1，如果为 1，查询的是逻辑帧数；如果为 0，查询的是渲染帧数。
两者之差可以得到无效帧数（被跳过渲染的帧数，仅在调用 `delayfps` 会产生）。
如果没有调用过 `delay_jfps`，那么两者无区别。

返回值：

返回当前刷新率。

说明：

FPS(Frames Per Second)：每秒传输帧数。通常，这个帧数在动画或者游戏里，至少要达到 30 才能基本流畅。现代液器均使用 60FPS 的刷新率，所以，如果你希望在你的显示器上达到最佳效果，那你需要至少 60FPS。

而使内部 FPS计数增加的方式是当你绘图后，调用 dy 族函数，如 :delay, delay_ms, delay_fps, Sleep ，否则你不调用这些函数时，FPS 永远为 0 而不会变化。

示例：

参见示例程序中的“ 星空 ”

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.9.3 GetHWND

HWNDGetHWND();

参数：

(无)

返回值：

返回绘图窗口句柄。

说明：

在 Windows 下，句柄是一个窗口的标识，得到句柄后，可以使用 Wows SDK 中的各种命令实现对窗口的控制。

示例：

```
// 获得窗口句柄
HWNDhWnd = GetHWND();
// 使用 API 函数修改窗口名称
SetWindowText (hWnd, TEXT("Hello!"));
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.9.4 inputbox getline

```
int inputbox_getline ( LPCSTR title, LPCSTR text, LPSTR buf, int len);
int inputbox_getline ( LPCWSTR title, LPCWSTR text, LPWSTR buf, int len);
```

参数：

title

对话框标题

text

对话框内显示的提示字，可以使用 `\n` 或者 `\t` 进行格式控制。

buf

用于接收输入的字符串指针，指向一个缓冲区

len

指定 buf 指向的缓冲区大小，同时也会限制在对话框里输入的最大长度

返回值：

返回 1 表示输入有效，buf 中的内容为用户所输入的数据，返回 0 表示输入无效，同时 buf 清空。

示例：

```
#include "graphics.h"
```

```
int main()
```

```
{
```

```
    initgraph (640, 480);
```

```
    char str[100];
```

```
    inputbox_getline ("这是一个对话框",
                      "请随便输入一串字符，输入完请回车",
```

```
                      str,
```

```
                      sizeof (str)/ sizeof (*str));
```

```
    ttextxy (0, 0, str);
```

```
    getch ();
```

```
    return 0;
```

```
}
```

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.10 随机函数

5.10.1 随机函数

[主页 \(See 1.\)](#)-> [库函数目录 \(See 5.1\)](#)-> 随机函数

目录：

函数或数据	说明
random (See 5.10.2)	生成 0 - n-1 范围内的随机数
randomf (See 5.10.3)	生成 0.0-1.0 范围内的随机数 (0.0 取到 , 1.0 取不到)
randomize (See 5.10.4)	初始化随机数序列

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.10.2 random

unsigned int random(unsigned int n = 0);

参数：

n

生成 0 至 n-1 之间的整数。

如果 n 为 0，则返回 0 - 0xFFFFFFFF 的整数。

返回值：

返回一个随机整数。

其它说明：

建议不要使用 `stdlib` 里的 `rand` 函数，而改用本函数。本函数使用专业的数生成算法，随机性远超系统的 `rand` 函数。另，千万不要自己 `random() % n` 的方式获得一个范围内的随机数，请使用 `random(n)`，切记。本列的初始化只能调用 `randomize` 函数，不能使用 `srand`。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.10.3 randomf

```
double randomf ();
```

参数：

无

返回值：

返回一个 01 之间的随机浮点数， 0.0 可能取得到， 1.0 一定取不到。

其它说明：

本随机序列的初始化只能调用 `randomize` 函数能使用 `srand` 。

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.10.4 randomize

[主页](#) (See 1.)- > [库函数目录](#) (See 5.1)- > [随机函数](#) (See 5.10.1)- >

`randomize`

功能：

这个函数用于初始化随机数序列。如果不调用本函数，那么 `random` 返回的序列将会是确定不变的。

声明：

```
void randomize ();
```

参数：

(无)

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11 时间函数

5.11.1 时间函数

[主页 \(See 1.\)](#)-> [库函数目录 \(See 5.1\)](#)-> 时间函数

目录：

函数或数	说明
api_sleep(See 5.11.2)	实际调用 API 中的 Sleep，因直接调用 Sleep 会被转化为调用 delay
delay(See 5.11.3)	至少延迟以毫秒为单位的时间
delay_ms(See 5.11.6)	平均延迟以毫秒为单位的时间
delay_fps(See 5.11.4)	延迟以 FPS为准的时间，以实现稳定帧
delay_jfps(See 5.11.5)	延迟以 FPS为准的时间，以实现稳定帧率（带跳帧）
fclock(See 5.11.7)	获取当前程序从初始化起经过的时间，以秒为单位

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.2 api sleep

参数：

dwMilliseconds
要延迟，以毫秒为单位，如果为 0 则不产生延时的作用（相当于无意义调用）。
不会附带刷新窗口的作用。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.3 delay

```
void delay (long milliseconds);
```

参数：

Milliseconds

延迟的时间，以毫秒为单位

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.4 delay_fps

```
void delay_fps (long fps);
```

```
void delay_fps (double fps);
```

参数：

fps

要得到的帧率，平均延迟 $1000/\text{fps}$ 毫秒，并更新 FPS计数值函数一秒最多能调用 fps 次。

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.5 delay jfps

参数：

fps

要得到的帧率，平均延迟 $1000/\text{fps}$ 毫秒，并更新 FPS计数值。这个函数一秒最多能调用 fps 次。注意的是，即使这帧跳过了，仍然 FPS计数值。

值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.6 delay ms

`void delay_ms (long Milliseconds);`

参数：

Milliseconds

要延迟的时间，以毫秒为单位

返回值：

(无)

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

5.11.7 fclock

double fclock ();

参数：

(无)

返回值：

返回一个以秒为单位的浮点数，精度比 API 的 GetTickCoun 高。程序中使用一般用于求时间差，一般不要直接使用这个值。

示例：

(无)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6. 入门基础示例

6.1 入门基础示例

[主页 \(See 1.\)](#)- > 入门基础示例

绘图入门部分目录（建议以下内容使用安装包里的 /man/tutorial/index.htm 来浏览，若在线浏览可能会乱码）：

例子名字	说明
第一个图形窗口程序 (See 6.3)	第一个图形窗口程序，学会建立窗口界面
第一个画图程序 (See 6.2)	学会绘画简单的图形，以及基本概念介绍说明
色彩的计算 (See 6.18)	学会使用指定绘图颜色，以及更多的画图函数

颜色填充 (See 6.25)	使用填充颜色，画实心图案
图形边界和填充 (See 6.22)	分别设置图形边界颜色和填充颜色来绘图
屏幕背景色 (See 6.17)	设置屏幕的背景色
文字绘制与文字相关设置 (See 6.24)	文字绘制与文字相关设置
图形视口裁剪 (See 6.23)	图形视口裁剪
使用 PIMAGE——抓图，绘画到 PIMAGE，PIMAGE 到屏幕 (See 6.21)	PIMAGE 的基本使用方法
使用 PIMAGE——图片缩放 (See 6.20)	使用 PIMAGE——图片缩放
使用 PIMAGE——透明，半透明 (See 6.19)	使用 PIMAGE——透明，半透明
用户交互 —— 键盘 1(See 6.26)	用户交互 —— 键盘
用户交互 —— 键盘 2(See 6.27)	用户交互 —— 键盘
用户交互 —— 鼠标 1(See 6.28)	用户交互 —— 鼠标 1
用户交互 —— 鼠标 2(See 6.29)	用户交互 —— 鼠标 2
用户交互 —— 字符串数据输入 (See 6.30)	用户交互 —— 字符串数据输入
更多的文字对齐方式设置 (See 6.5)	更多的文字对齐方式设置
更多的颜色表示办法 (HSV/HSL)(See 6.6)	更多的颜色表示办法 (HSV/HSL)
高级：绘画渐变色 /彩虹 (See 6.4)	高级：绘画渐变色 /彩虹

图形动画入门部分目录

例子名字	说明
基础动画一：规范框架 (See 6.16)	第一个图形动画规范框架示例
基础动画二：简单平移动画 (See 6.8)	简单平移示例
基础动画三，更丰富的变化与帧率 (See 6.12)	碰撞加帧率显示示例
基础动画四，半透明淡入淡出 (See 6.14)	透明半透明控制显示示例
基础动画五，对象封装，使代码更有条理 (See 6.15)	对象封装示例
基础动画六，利用对象封装，显示多个对象 (See 6.10)	对象封装应用示例
基础动画七，使用非整数参数，随机色彩，使你的动画更自然 (See 6.11)	对象封装复杂应用示例
基础动画八，使用 C++封装你的对象 (See 6.7)	C++对象封装示例
基础动画九，使用场景，更灵活地控制动画 (See 6.9)	场景控制示例
基础动画十，场景管理下的输入输出控制 (See 6.13)	场景输入输出控制示例 （事实上已经是简单的游戏示例）

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.2 第一个画图程序

[主页 \(See 1.\)](#) - > [入门基础示例 \(See 6.1\)](#) - > 第一个画图程序

里，介绍一下坐标的概念。

为了能方便描述一个点的位置，我们需要使用坐标。

以窗口的左上角为基准，某个点，距离左边界的距离，我们记录为 x 值；距离上边界的距离，我们记录为 y 值。

那么可以用一对 (x, y) 来表示这个点，这个点距离上边为 x ，距离左边为 y ，这样就知道这个点的位置了。

这个和数学里的坐标概念是一样的，只是方向与数学上的不相同。

那么，这个距离，我们使用什么单位呢？在这里我们使用“像素”为单位。

所谓像素（pixel），你留意一下屏幕，对它放大很多倍的话，你会发现它是由很多个小正方形色块构成的，

而这些小正方形，一个就是一像素，像素就是图像的最基本的单位。

在新建立的时候，那个尺寸所用的单位也是像素。比如你建立了一个 640×480 的窗口，

那么最左上角的像素的坐标是 $(0, 0)$

最右下角的坐标是 $(639, 9)$

注意不是 $(640, 480)$ ，是要减 1 的。

若显示乱码请 [点此下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.3 第一个图形窗口程序

[主页 \(See 1.\)](#) - > [入门基础示例 \(See 6.1\)](#) - > 第一个图形窗口程序

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by

TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.4 高级：绘画渐变色 / 彩虹

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 高级：绘画渐变色 / 彩虹

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.5 更多的文字对齐方式设置

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 更多的文字对齐方式设置

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.6 更多的颜色表示办法 (HSV/HSL)

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 更多的颜色表示办法

(HSV/HSL)

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.7 基础动画八，使用 C++封装你的对象

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画八，使用 C++封装你的对象

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.8 基础动画二：简单平移动画

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画二：简单平移动画

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.9 基础动画九，使用场景，更灵活地控制动画

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画九，使用场景，更灵活地控制动画

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.10 基础动画六，利用对象封装，显示多个对象

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画六，利用对象封装，显示多个对象

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.11 基础动画七，使用非整数参数，随机色彩，使你的动画更自然

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画七，使用非整数参数，随机色彩，使你的动画更自然

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.12 基础动画三，更丰富的变化与帧率

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画三，更丰富的变化与帧率

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.13 基础动画十，场景管理下的输入输出控制

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画十，场景管理下的输入输出控制

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.14 基础动画四，半透明淡入淡出

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画四，半透明淡入淡出

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.15 基础动画五，对象封装，使代码更有条理

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画五，对象封装，使代码更有条理

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.16 基础动画一：规范框架

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 基础动画一范框架

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.17 屏幕背景色

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 屏幕背景色

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.18 色彩的计算

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 色彩的计算

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.19 使用 IMAGE ——透明，半透明

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 使用 IMAGE——透明，半透

明

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.20 使用 IMAGE——图片缩放

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 使用 IMAGE——图片缩放

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.21 使用 IMAGE——抓图，绘画

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 使用 IMAGE——抓图，绘画

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.22 图形边界和填充

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 图形边界和填充

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.23 图形视口裁剪

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 图形视口裁剪

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.24 文字绘制与文字相关设置

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 文字绘制与文字相关设置

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.25 颜色填充

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 颜色填充

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.26 用户交互——键盘

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 用户交互——键盘 1

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.27 用户交互——鼠标 1

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 用户——键盘 2

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.28 用户交互——鼠标 1

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 用户交互——鼠标 1

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.29 用户交互——鼠标 2

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 用户交互——鼠标 2

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

6.30 用户交互——字符串数据输入

[主页 \(See 1.\)](#)- > [入门基础示例 \(See 6.1\)](#)- > 用户交互——字符串数据输

入

若显示乱码请 [点此打开下载](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

7. 示例程序

7.1 ege 示例程序

[主页](#) (See 1.)- > ege 示例程序

以下为已编译好的示例程序

更多的演示程序及其可以在 [ege 安装包内 demo目录](#)找到：

简单的演示

[五角星动画](#)

[星空屏保动画](#)

[鼠标弹球动画](#)

略为复杂的演示

[变幻线屏保](#)

[彩色三角形屏保](#)

[时钟（带抗锯齿演示）](#)

[烟花特效](#)

[弹球 1](#)

[弹球 2（抗锯齿版本）](#)

[碧波荡漾](#)

完整游戏演示

[俄罗斯方块](#)

[中国象棋](#)

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

8. 网友程序

8.1 ege 网友小程序

[主页 \(See 1.\)](#)- > eg 小程序

以下为示例程序源代码

更多的演示程序及其源代码可以在 ege 安装包内 demo 目录找到：

简单示

[分形叶子——菊花老怪](#)

[小球弹幕——菊花老怪](#)

[弹幕演示—— doufuguolyl \(豆腐果\)](#)

[直线和球的碰撞检测—— wysaid](#)

[迷宫求解—— wysaid](#)

小游戏

拼图游戏——忆_碎碎念

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.

9. 关于本 EGE图形库

9.1 关于本 EGE 图形库

主页 (See 1.)- > 关于本 EGE图形库

相关信息：

作者：misakamm

Blog：<http://log.misakamm.org>

Mail：misakamm[at]gmail[dot]com

更新历史：

13.03

画圆 / 椭圆 / 弧函数增加浮点版本

填充函数增加按颜色区域填充（类似画图的油漆桶功能）

修正 IMAGE初始化背景色不色

修正第二次 initgraph 的状态错误

源代码结构细调，抽离数据结构实现

13.02

图形效率优化

getch 特殊键响应不正常修正

修正 keystate 状态错误

.01

对 mouse_msg友好化

12.12

增加 putimage_withalpha

修正 ege_puttexture 贴图 bu

12.11

增加对 VC2012 的支持

发布文件分类方式调整

12.08

delay_fps/delay_jfps 增加浮点帧率支持

修正文档已知错误

12.06

新增函数 xyprintf/rectprintf

修正 png 处理 alpha 为 0 的像素的方案

修正对带 alpha 的图像的贴图支持

增加离线示例和在线示例

vs2010 编译错误修正

rgb2gray 函数修正

12.05

inputbox_getline 函数修正

mousemove消息参数修正

增加发布包中的示例

12.04

库函数调整较大，去除所有大写字母开头的函数，旧程序基本都需要改动才能正常编译

mingw 编译参数有变，增加链 simg32

不允许使用 IMAGE, 全部改用 PIMAGE, 使用 newimage/delimage 构造和删除
颜色全部统一为 ARGB格式，不要使用宏 R 要用 EGERG代替（否则红蓝颠倒）

增加带 aa 的绘图

增加渐变填充

增加纹理概念，增加简单的纹理贴图

修正若干 bugs（主要的是鼠标移动消不正确）

文档增加动画示例

增加支持 64 位静态库（仅 vc9/vc10 ）

去除原有的编译警告

增加部分函数过时警告，不屏蔽 vs 的过时警

示例代码更新及增加

增加邮件列表通讯讨论

<https://lists.sourceforge.net/lists/listinfo/tcgraphics-qs>

12.03 stable 稳定版 svn89

调整版本命名方式 (年 . 月 , 像 12.03 即 12 年三月发布)

调整编译参数

默认标题风格调整

源代码目录结构调整

2012 年 :

0107 :

对 objc++ 编译做兼容

部分函数风格调整

增加控件 button (未完善 , 测试版)

2011 年 :

0724 :

Bug 修正

解决在 vs2010 下编译不正确的问题

fix 在 linux wine 下窗口移动困难的问题

0602 :

增加 GUI

修正部分 png 读取错误

调整 getkey 函数 , 增加配对的 kbmsg 函数

增加 getmouse 函数

增加 saption 函数

版本更新为 20110602 V1.0.1 Beta

0520 :

调整头文件 , 以使兼容性更好 , 减少链接出错

GetFPS 掉参数

版本更新为 20110520 V0.3.8 Release

0416 :

增加 png 图片格式的读写读取仍然用 getimage , 保存用 savepng)

增加压缩解压函数 (ege_compress/ege_uncompress/ege_uncompressze)

版本更新为 20110416 V0.3.8 Beta

0414 :

修正拖动时 , 窗口可能会漂移的问题

重绘效率优化

增加 INIT_LOCKWINDOW

批量绘图函数改名为窗口锁定函数 (WindowLock/WindowUpdate/WindowUnlock),
原函数仍然能使用

IMAGE类增加 getdc 接口以便 GDI 操作 (仅可用于 Blt 操作其它操作慎用)

VC6库改用 /MT 编译以便多线程编程

版本更新为 20110414 V0.3.7 Release

0401 :

增加 drawlines 和 drawbezier 函数, 优化 drawpoly 的效率

增加对设幕保护程序的支持

增加安装程序

增加 WM_MOUSEWH 宏定义以防在 VC6上编译出错

文档更新

版本更新为 20110401 V0.3. Release

0321 :

更名为 EGE, 取代原 EGE(Easy Graphics Engine) 库

增加旋转缩放贴图函数 (接口还可能变化, 暂使用)

增加窗口尺寸自动修正 (在分辨率变小强制窗口尺寸变小了, 再变回大分辨率的时候)

版本更新为 20110321 V0.3.5 Release

29 :

修正头文件一个类型错误的问题

增加 saveimage 函数, 用来保存图片 (目前只支持 bmp)

版本号更新为 20110129 V0. Release

2010 年以前的更新记录 :

1213 :

增加对基本音频播放的支持 (wav, mp3, midi)

版本号更新为 20101213 V0.3.3 Release

1212 :

更新文档

增加独立下载库和帮助文档

版本号更新为 20101207 V0.3.2 Rese

1207 :

修正图形更新同步时可能偶然发生无法更新的情况

增加 setbkcolor_f 函数, 功能是仅设置背景颜色,

仅当 cleardevice 时才使用这个颜色一次性清除

提升了重绘性能, 并修正一个资源泄露的问题

增加 swappage 函数, 实现双缓冲模式

1122 :

调整 GetMousePos 初始化坐标
 鼠标滚轮坐标不正确的问题
 随机函数改用专业算法，并增加 randomf 函数
 增加 3D 数学函数
 增加 3D 向量类
 修正文档错误，补充没加到文档的函数
 版本号改为 20101122 V0.3.1 Release

1118 :

修正了三个新贴图函数的 BUG
 增加了 outtextrect, setfontbkcolor
 增加了 logo 显示
 修正鼠标消息接收时内容不全的问题
 修正多次开关批量绘图时可能会卡死的 BUG
 相应更新文档
 版本号改为 20101118 V0.3.0 Release

1113 :

对 imagefilter_blurring 函数调整参数次序，并增加默认参数其使用更方便
 优化透明，半透明贴图效率

1111 :

增加三个函数支持透明 / 半透明贴图 (putimage_transparent, putimaalphablend, putimage_alphatransparent)
 增加图像模糊函数 (imagefilter_blurring)
 版本号改为 20101111 V0.2.5 Gamma

1101 :

调整 putpixels ，使得效率更高一些
 修正部分绘图函数如 rangle 不空心的 BUG
 增加 INITGRAPH宏，以便于使用全局 IMAGE时，抢先初始化
 相应更新文档
 版本号改为 20101101 V0.2.4 Gamma

1023 :

增加 putpixels 函数，便于批量画点时提升速度
 修正 setbkcolor 函数颜色不正确的 BUG

1013 :

增加 GetHInstance 函数，以便作为读入资源时的参数
 调整 closegraph ，以便 close 后再次 init
 修正 setviewport 在某个版本里误编辑的问题导致区域失效
 xtwidht 和 textheight 增加单字符版本

修正总是隐藏控制台无法故意显示的问题

增加 KEYMSG_FIRSTDOWN 便于实现按键按下一次仅次的需求

修正相应的俄罗斯方块例子（因增加了 KEYMSG_FIRSTDOWN 的原因）

1010 :

增加 ShowMouse 函数，以控制鼠标是否隐

增加对 VC2005 的支持（不带 SP1，打了 SP1 的不能编译通过，其实为了更稳定建议直接用 VC2008）

增加 IMAGE 对象有效性检查，以便更容易发现指误

相应更新文档，并修正文档中部分已发现的问题

调整 initgraph，允许二次调用，并修改窗口大小（这时原内容，但除了 size 和 viewport 以外，其它属性不变）

修正批量画图多次调用时可能发生的死锁问题

增加初始化 IMAGE 在 initgraph 之前会直接报错并强制关闭程序

版本号改为 20101010 V0.2.2 Gamma

0930 :

调整 delay_fps，使得帧数更稳定些

修正 getch, kbhit 带参数时的一些问题

改为使用 SHOW_CONSOLE 显示控制台窗口，默认不显示，详见 PDF 内“超简单的使用预览”

增加 getpixel, putpixel_f，为原函数的加速版（但调用时有风险，但其实原来的已经足够快，现在的已经快不了多少）

增加 API_Sleep 函数用于调用原来的原版 S 函数

增加 delay_jfps 用于自动控制跳帧（跳多少看流畅程度，如果绘图不复杂计算量少，一般不会跳）

GetFPS 函数增加一个可选参数，以便得到实帧率（在跳帧模式下）

增加一些方便在 CFree 下编程的特性，如 WinMain，详见 PDF 内“超简单的使用预览”

基本完成 pdf，并加入了几个比较好的代码，包含一个俄罗斯方块程序

版本名改为 20101001 V0.2 Gamma

0928 :

前一次修改后对效率的调整和优化

修正 InputxGetLine, getch, GetMouseMsg 的 BUG

调整延迟函数的逻辑，使得在相同帧率下能更流畅

增加 fclock 函数

增加 mingw3.4.5 编译器使用的静态库，即能支持在 CFree 下编译

增加 pdf 格式的说明文档（测试版，部分函数还没有补充完成）

版本名改为 0.2 Beta

0926 :

大幅修改，使程序不会因为鼠标拖动而停下

即使不调用 `delay` 族函数也会更新窗口，只是并不流畅，所以如果你想要流畅动画，推荐手工调用 `delay` 族函数

增加 `GetFPS` 函数，可以随时获取当前帧率

`GetLineInputBox` 更名为 `InputBoxGetLine`，以便于增加类似 `InputGetNumber` 时，相同前缀容易查找

`getch` 和 `kbhit` 改为宏，以防与 `conio.h` 库冲突，当包含 `conio.h` 时，会自动忽略

版本名改为 0. Alpha

注：`delay` 族函数在 `graphics.h` 头文件的最顶处有相关的说明，就是指 `Sleep`, `delay`, `delay_ms`, `delays`, `getch`, `GetMouseMsg`

0923：

修正 `getch` 的行为，以保持非特殊按键键码与 CRT(C Run-Time) 的 `getch` 一致

0922：

增加 `GetLineInputBox` 在图形界面下进行字符串输入，使用回车确认输入，ESC 取消输入（输入为空）

扩展 `getch` 和 `kbhi`，使得能根据参数把按键放开的事件也能取得，两个函数需要组合使用

0921：

增加 `void setinitmode(int mode, int x = CW_USEDEFAULT, int y = CW_USEDEFAULT)`

第一个参数 `mode=0` 为普通窗口，`1` 为无边框窗口，`xy` 是初始窗口坐标，`mode` 为 `1` 时，用于方便全屏模式

此函数必须在 `initgraph` 前调用，否则无效

`initgraph` 函数，若窗口大小均指定为 `-1,-1`，则使用屏幕大小，用于方便全屏后可用 `getwidth` 和 `getheight` 取得窗口大小

VC下使用 `HIDE_CONSOLE` 可以使控制台窗口彻底消失，使用方法如下：

```
#define DE_CONSOLE
```

```
#include "graphics.h"
```

要注意的是，`#define HIDE_CONSOLE` 必须在 `graphics.h` 的前面为 Beta 版本

0920：

修正若干 BUG

0918：

第一版发布测试

Caution: Some letters were intendedly removed from the document because It was created by TRIAL version of Softany CHM to PDF converter. To get rid of this, please purchase the product.