



Microsoft Windows 3.0

程序设计指南

北京科海培训中心

Microsoft Windows 3.0

程序设计指南

北京科海培训中心

目 录

绪论	(1)
在开始前首先应了解什么?	(1)
本指南的内容	(1)
必须具备的工具	(2)
使用应用程序实例	(3)
符号约定	(3)

第一部分 如何写Windows应用程序

第一章 Windows环境概述	(5)
1.1 Microsoft Windows与DOS的比较	(5)
1.1.1 用户界面	(6)
1.1.2 队列输入	(6)
1.1.3 与设备无关的图形	(7)
1.1.4 多任务	(7)
1.2 Windows程序设计模式	(7)
1.2.1 窗口	(8)
1.2.2 菜单	(8)
1.2.3 会话框	(8)
1.2.4 消息循环	(9)
1.3 Windows库	(9)
1.4 建立Windows应用程序	(11)
1.5 软件开发工具	(12)
1.5.1 C编译程序	(12)
1.5.2 连接程序	(12)
1.5.3 SDK资源编辑程序	(13)
1.5.4 资源编译程序	(13)
1.5.5 调试与优化工具	(13)
1.5.6 程序维护程序	(14)
1.6 编写Windows应用程序的注意事项	(15)
1.7 小结	(16)
第二章 一个典型的Windows应用程序	(17)

2.1	应用程序Generic	(17)
2.2	Windows应用程序.....	(17)
2.3	WinMain函数.....	(18)
2.3.1	Windows中的数据类型与结构.....	(18)
2.3.2	句柄.....	(19)
2.3.3	事例.....	(19)
2.3.4	登录窗口类.....	(20)
2.3.5	创建窗口.....	(22)
2.3.6	显示与更改窗口.....	(23)
2.3.7	建立消息循环.....	(24)
2.3.8	产生控制.....	(25)
2.3.9	终止应用程序.....	(25)
2.3.10	初始化函数	(26)
2.3.11	应用程序命令行参数	(27)
2.4	窗口函数.....	(27)
2.5	建立About (询问) 会话框	(29)
2.5.1	建立会话框样板.....	(29)
2.5.2	建立标题文件.....	(30)
2.5.3	建立会话函数.....	(31)
2.5.4	定义带About命令的菜单	(32)
2.5.5	处理消息WM_COMMAND	(32)
2.6	建立模块定义文件.....	(33)
2.7	组织Generic	(35)
2.7.1	建立C语言源文件	(35)
2.7.2	建立标题文件.....	(41)
2.7.3	建立资源描述文件.....	(41)
2.7.4	建立模块定义文件.....	(41)
2.7.5	建立Make (维护) 文件	(42)
2.7.6	运行MAKE程序.....	(43)
2.8	用Generic作为样板	(43)
2.9	小结.....	(44)

第二部分 Windows应用程序设计

第三章	窗口输出	(45)
3.1	显示上下文.....	(45)
3.1.1	调用GetDC函数.....	(46)
3.1.2	WM_PAINT消息	(46)

3.1.3	屏蔽用户区域.....	(47)
3.1.4	显示上下文和设备上下文.....	(48)
3.1.5	座标系.....	(48)
3.2	作图工具的创建、选择和删除.....	(48)
3.3	作图和写字.....	(49)
3.4	一个应用程序实例：Output.....	(51)
3.4.1	增加新变量.....	(51)
3.4.2	增加WM_CREATE Case 语句.....	(51)
3.4.3	增加WM_PAINT Case 语句.....	(52)
3.4.4	修改WM_DESTROY Case 语句.....	(55)
3.4.5	编译与连接.....	(55)
3.5	小结.....	(56)
第四章	键盘与鼠标器输入	(57)
4.1	Windows输入信息.....	(57)
4.1.1	消息格式.....	(57)
4.1.2	键盘输入.....	(58)
4.1.3	字符输入.....	(58)
4.1.4	鼠标器输入.....	(58)
4.1.5	定时器输入.....	(59)
4.1.6	滚动杆输入.....	(60)
4.1.7	菜单输入.....	(60)
4.2	一个应用程序实例：Input	(61)
4.2.1	应用程序Input如何显示输出	(61)
4.2.2	增加新变量.....	(62)
4.2.3	设置窗口类格式.....	(63)
4.2.4	修改Create Window函数.....	(63)
4.2.5	设置文本矩形.....	(63)
4.2.6	增加WM_CREATE Case 语句.....	(64)
4.2.7	修改WM_DESTROY Case 语句.....	(64)
4.2.8	增加WM_KEYUP和WM_KEYDOWN Case语句.....	(64)
4.2.9	增加WM_CHAR Case语句	(64)
4.2.10	增加WM_MOUSEMOVE Case语句	(65)
4.2.11	增加WM_LBUTTONDOWN和WM_LBUTTONDOWN Case语句.....	(65)
4.2.12	增加WM_LBUTTONDOWNCLK Case语句	(65)
4.2.13	增加WM_TIMER Case语句	(65)
4.2.14	增加WM_HSCROLL和WM_VSCROLL Case语句.....	(66)
4.2.15	增加WM_PAINT Case语句	(66)
4.2.16	编译与连接	(67)

4.3 小结	(67)
第五章 图标 (icon)	(68)
5.1 什么是图标	(68)
5.1.1 使用内部图标	(69)
5.2 使用自定义的图标	(69)
5.2.1 创建一个图标文件	(69)
5.2.2 定义图标资源	(69)
5.2.3 装入图标资源	(69)
5.3 指定一个类图标	(70)
5.4 显示自定义的图标	(70)
5.5 在对话框中显示一个图标	(71)
5.6 应用程序实例: Icon	(72)
5.6.1 增加一个ICON语句	(72)
5.6.2 增加一条ICON控制语句	(72)
5.6.3 设置类图标	(73)
5.6.4 在Make文件中增加MYICON.ICO	(73)
5.6.5 编译与连接	(73)
5.7 小结	(73)
第六章 光标、鼠标器与键盘	(74)
6.1 控制光标的形状	(74)
6.1.1 使用内部光标形状	(74)
6.1.2 使用自定义的光标形状	(74)
6.2 显示光标	(75)
6.2.1 指定类光标	(75)
6.2.2 显式设置光标形状	(75)
6.2.3 例子: 显示一个费时操作的沙漏	(76)
6.3 用户用鼠标选择信息	(77)
6.3.1 开始一个图形选择	(77)
6.3.2 显示选择	(79)
6.3.3 结束选择	(80)
6.4 利用键盘移动光标	(81)
6.4.1 利用键盘移动光标	(81)
6.4.2 当无鼠标器时使用光标	(83)
6.5 一个应用程序例子: Cursor	(84)
6.5.1 增加CURSOR语句	(84)
6.5.2 增加新变量	(84)
6.5.3 设置类光标	(85)

6.5.4	准备沙漏光标	(85)
6.5.5	增加一个费时操作	(85)
6.5.6	增加WM_LBUTTONDOWN、WM_MOUSE_MOVE和WM_LBUTTONUP Case语句	(86)
6.5.7	增加WM_KEYDOWN和WM_KEYUP Case语句	(88)
6.5.8	增加WM_PAINT Case语句	(89)
6.5.9	在Make文件中增加BULLSEYE.CUR	(89)
6.5.10	编译和连接	(89)
6.6	小结	(90)

第七章	菜单	(91)
7.1	菜单是什么?	(91)
7.2	定义菜单	(91)
7.2.1	菜单ID	(92)
7.3	在你的应用程序中包含一个菜单	(93)
7.3.1	为一个窗口类指定一个菜单	(93)
7.3.2	为一个特定窗口指定一个菜单	(93)
7.4	处理从菜单输入的信息	(94)
7.5	应用程序用菜单工作	(95)
7.5.1	打开和关掉菜单项	(95)
7.5.2	检验与非检验菜单项	(96)
7.5.3	增加菜单项	(97)
7.5.4	修改已存在的菜单	(98)
7.5.5	删除菜单项	(98)
7.5.6	位图用作菜单项	(99)
7.5.7	替换菜单	(99)
7.5.8	创建新菜单	(100)
7.5.9	初始化菜单	(101)
7.6	特殊菜单特征	(101)
7.6.1	提供菜单加速键	(101)
7.6.2	使用级联式菜单	(104)
7.6.3	使用浮动弹出型菜单	(105)
7.6.4	设计你自己的检验标志	(106)
7.6.5	使用所有主画图菜单	(107)
7.7	一个应用程序例子: EditMenu	(108)
7.7.1	在资源文件中增加新菜单	(109)
7.7.2	在包含文件中增加定义	(110)
7.7.3	在资源描述文件中增加加速表	(110)
7.7.4	增加一个新变量	(111)

7.7.5	装载加速表	(111)
7.7.6	修改消息循环	(111)
7.7.7	修改WM_COMMAND Case语句	(111)
7.7.8	编译和连接	(112)
7.8	小结	(112)
第八章	控制	(113)
8.1	控制是什么?	(113)
8.2	建立一个控制	(113)
8.2.1	指定控制类	(114)
8.2.2	选择控制形式	(114)
8.2.3	设置父窗口	(115)
8.2.4	选择一个控制ID	(115)
8.3	使用控制	(115)
8.3.1	接收用户输入	(116)
8.3.2	传送控制消息	(116)
8.3.3	允许或禁止输入到控制	(116)
8.3.4	对控制移动和改变大小	(116)
8.3.5	删除一个控制	(117)
8.4	建立和使用一些普通控制	(117)
8.4.1	按钮控制	(117)
8.4.2	静态控制	(120)
8.4.3	列表框	(121)
8.4.4	组合框	(127)
8.4.5	编辑控制	(128)
8.4.6	滚动杆 (Scroll bar)	(130)
8.5	应用实例程序: Edit Cntl	(131)
8.5.1	在包含文件中增加一个新常量	(132)
8.5.2	增加新变量	(132)
8.5.3	加入Greate Window函数	(132)
8.5.4	修改WM_COMMAND case语句	(133)
8.5.5	加入WM_SETFOCUS case语句	(133)
8.5.6	增加WM_SIZE	(133)
8.5.7	编译和连接	(134)
8.6	小结	(134)
第九章	会话框	(135)
9.1	什么是会话框	(135)
9.1.1	形式会话框	(135)

9.1.2	非形式会话框	(136)
9.2	使用会话框	(136)
9.2.1	建立会话函数	(137)
9.2.2	在会话框内使用控制	(138)
9.3	一个样本应用程序: FileOpen	(138)
9.3.1	对包含文件增加常量	(139)
9.3.2	建立Open会话框样板	(139)
9.3.3	增加新的变量	(140)
9.3.4	增加IDM_OPEN情况语句	(140)
9.3.5	建立OpenDlg函数	(141)
9.3.6	增加帮助程序函数	(143)
9.3.7	输出会话函数	(145)
9.3.8	编辑和连接	(145)
9.4	小结	(145)
第十章	文件的输入/输出	(146)
10.1	处理Windows环境中文件的规则	(146)
10.2	建立文件	(148)
10.3	打开现存文件	(148)
10.4	从文件中读取以及向文件中写入	(148)
10.5	重新打开文件	(149)
10.6	文件提示	(149)
10.7	检查文件状态	(150)
10.8	一个简单的文件编辑程序: EditFile	(150)
10.8.1	在包含文件中增加一个常量	(150)
10.8.2	增加SaveAs会话框	(150)
10.8.3	增加包含语句	(151)
10.8.4	增加新的变量	(151)
10.8.5	更换WM_COMMAND case语句	(152)
10.8.6	增加WM_QUERYEND_SESSION和WM_CLOSE case语句	(154)
10.8.7	修改OpenDlg会话函数	(154)
10.8.8	增加SaveAsDlg会话函数	(155)
10.8.9	增加帮助函数	(156)
10.8.10	输出SaveAsDlg会话函数	(159)
10.8.11	为局部堆增加空间	(159)
10.8.12	编译和连接	(159)
10.9	小结	(160)
第十一章	位图	(161)

11.1	什么是位图?	(161)
11.2	创建位图	(161)
11.2.1	创建和装载位图文件	(162)
11.2.2	创建和填充一个空白位图	(162)
11.2.3	使用硬编码位创建位图	(163)
11.2.4	绘制彩色位图	(165)
11.3	显示位图	(166)
11.3.1	使用BitBlt函数显示一个内存位图	(167)
11.3.2	放大位图	(168)
11.3.3	在模式电刷中使用位图	(169)
11.3.4	显示一个于设备无关的位图	(170)
11.3.5	位图作为菜单项	(171)
11.4	在单色位图中加入颜色	(171)
11.5	删除位图	(171)
11.6	一个应用程序实例: Bitmap	(172)
11.6.1	修改include文件	(172)
11.6.2	加入位图的源数据	(173)
11.6.3	加入位图、模式和方式菜单	(173)
11.6.4	加入全局和局部变量	(173)
11.6.5	增加WM_CREATE Case语句	(174)
11.6.6	修改WM_DESTROY Case语句	(177)
11.6.7	增加WM_LBUTTONDOWN, WM_MOUSEMOVE和WM_LBUTTONDOWN Case语句	(177)
11.6.8	增加WM_RBUTTONDOWN Case语句	(178)
11.6.9	增加WM_ERASEBKGD Case语句	(178)
11.6.10	修改WM_COMMAND Case语句	(179)
11.6.11	修改Make file	(180)
11.6.12	编译和连接	(181)
11.7	小结	(181)
第十二章	打印	(182)
12.1	在Windows环境中打印	(182)
12.1.1	使用打印机换码符	(182)
12.2	搜索当前打印机的有关信息	(183)
12.3	打印一行正文	(184)
12.4	打印一个位图	(185)
12.5	处理打印过程中的错误	(186)
12.6	取消一个打印操作	(187)
12.6.1	定义一个Abort对话框	(188)

12.6.2	定义一个Abort对话框函数	(188)
12.6.3	定义一个Abort函数	(189)
12.6.4	终止打印操作	(190)
12.6.5	使用ABORTDOC换码序列取消一个打印操作	(190)
12.7	使用频带来打印图象	(190)
12.8	一个样本应用程序: PrintFile	(191)
12.8.1	增加一个AbortDlg对话框	(192)
12.8.2	为打印增加变量	(192)
12.8.3	增加IDM_PRINT情况语句	(193)
12.8.4	生成AbortDlg和AbortProc函数	(195)
12.8.5	增加GetPrinterDC函数	(195)
12.8.6	输出AbortDlg和AbortProc函数	(196)
12.8.7	编译和连接	(196)
12.9	小结	(196)
第十三章	书写板 (Clipboard)	(197)
13.1	使用书写板	(197)
13.1.1	往书写板上拷贝文本	(198)
13.1.2	在书写板上贴入文本	(199)
13.1.3	在书写板上贴入位图	(201)
13.1.4	Windows 书写板应用程序	(203)
13.2	使用特殊的书写板特性	(203)
13.2.1	根据需要再现数据	(204)
13.2.2	在终止前再现格式	(204)
13.2.3	记录私用格式	(204)
13.2.4	在书写板上控制数据显示	(205)
13.3	一个应用程序样本: ClipText	(206)
13.3.1	加入新的变量	(207)
13.3.2	修改例子初始化源码	(207)
13.3.3	加入一个WM_INITMENU情况语句	(207)
13.3.4	修改WM_COMMAND情况语句	(208)
13.3.5	增加WM_PAINT情况语句	(210)
13.3.6	增加OutOfMemory函数	(210)
13.3.7	编译和连接	(210)
13.4	小结	(211)

第三部分 高级程序设计

第十四章	C语言及汇编语言	(212)
-------------	-----------------------	--------------

14.1	选择内存模型	(212)
14.2	使用NULL	(213)
14.3	使用命令行参数和DOS环境	(213)
14.4	编写输出的函数	(214)
14.4.1	生成一个Callback 过程	(214)
14.4.2	生成函数WinMain	(215)
14.5	使用C运行库函数	(215)
14.5.1	使用Windows C库	(216)
14.5.2	分配内存	(216)
14.5.3	字符串操作	(216)
14.5.4	文件输入/输出的使用	(217)
14.5.5	使用控制台输入和输出	(218)
14.5.6	使用图形函数	(218)
14.5.7	使用浮点运算	(218)
14.5.8	执行其它的应用程序	(219)
14.5.9	使用BIOS和MS_DOS的接口函数	(219)
14.5.10	删除C运行库初始代码	(219)
14.6	编写汇编语言代码	(220)
14.6.1	修改中断标志	(222)
14.6.2	用汇编语言编写出口函数	(222)
14.6.3	使用ES寄存器	(223)
14.7	小结	(225)
第十五章	存储管理	(226)
15.1	内存的使用	(226)
15.1.1	使用全局堆	(226)
15.1.2	使用局部堆	(227)
15.1.3	可放弃存储块的使用	(228)
15.2	段的使用	(229)
15.2.1	使用代码段	(230)
15.2.2	DATA段	(231)
15.3	一个应用实例: Memory	(231)
15.3.1	分解C语言源文件	(231)
15.3.2	修改包含文件	(232)
15.3.3	增加新的段定义	(232)
15.3.4	修改Make文件	(233)
15.3.5	编译与连接	(233)
15.4	小结	(234)

第十六章 内存管理细节	(235)
16.1 Windows的内存配置	(235)
16.1.1 基本内存配置	(235)
16.1.2 EMS 4.0内存构置	(237)
16.1.3 Windows的标准内存配置	(240)
16.1.4 Window的386增强方式的内存配置	(243)
16.2 在Windows应用程序中使用数据存贮	(245)
16.2.1 管理自动数据段	(246)
16.2.2 管理局部动态数据块	(248)
16.2.3 管理全局内存块	(251)
16.2.4 使用窗口和类数据结构中的额外字节	(256)
16.2.5 资源管理	(257)
16.3 使用内存模型	(259)
16.4 使用巨型数据段	(260)
16.5 管理程序数据时应躲开的陷阱	(261)
16.6 如何管理程序代码的内存	(263)
16.6.1 使用代码段属性	(263)
16.6.2 使用多个代码段	(263)
16.6.3 代码段的平衡	(264)
16.6.4 .DEF文件中代码段的次序	(264)
16.7 小结	(265)
第十七章 打印设置	(266)
17.1 Windows如何管理打印设置	(266)
17.1.1 打印设置及DEVMODE结构	(267)
17.1.2 打印设置及打印机环境	(267)
17.2 使用设备驱动程序函数	(268)
17.3 查询打印机驱动程序的能力	(268)
17.4 使用打印设置	(268)
17.4.1 指定ExtDevice Mode的输入和输出	(269)
17.4.2 得到打印设置的副本	(270)
17.4.3 改变打印设置	(270)
17.4.4 用CreateDC函数来改变打印设置	(271)
17.4.5 改变打印设置而不影响其它应用程序	(273)
17.4.6 提示用户改变打印设置	(273)
17.5 驱动程序之间拷贝打印设置	(274)
17.6 维护你自己的打印设置	(274)
17.7 使用旧的打印机驱动程序	(275)

17.8	小结	(275)
第十八章	字体	(276)
18.1	文本输出	(276)
18.2	书写文本时使用颜色	(276)
18.3	使用备用字体	(277)
18.4	生成逻辑字体	(277)
18.5	在一行内使用多种字体	(278)
18.6	得到有关选定字体的信息	(279)
18.7	得到关于逻辑字体的信息	(280)
18.8	枚举字体	(281)
18.9	检查设备的文本能力	(282)
18.10	增加字体资源	(283)
18.11	设置文本对准	(284)
18.12	生成字体资源文件	(284)
18.12.1	生成字体文件	(285)
18.12.2	生成字体资源描述文件	(285)
18.12.3	生成伪码模块	(285)
18.12.4	生成模块定义文件	(286)
18.12.5	编译和连接字体资源文件	(287)
18.13	应用程序实例: ShowFont	(287)
18.14	小结	(287)
第十九章	彩色调色板	(288)
19.1	彩色调色板的功用	(288)
19.2	彩色调色板如何工作	(288)
19.3	生成并使用逻辑调色板	(290)
19.3.1	生成数据结构LOGPALETTE	(290)
19.3.2	生成逻辑调色板	(292)
19.3.3	将调色板选入设备上下文	(292)
19.3.4	实现调色板	(293)
19.4	用调色板彩色画图	(293)
19.4.1	直接指定调色板颜色	(293)
19.4.2	间接指定调色板颜色	(294)
19.4.3	使用调色板画位图	(295)
19.5	改变逻辑调色板	(295)
19.6	对系统调色板变化的响应	(297)
19.6.1	对WM_QUERYNEWPALETTE	(297)
19.6.2	对WM_PALETTECHANGED	(297)

19.7 小结	(298)
第二十章 动态连接库	(300)
20.1 什么是DLL?	(300)
20.1.1 输入库和DLL	(301)
20.1.2 DLL和应用模块	(301)
20.1.3 DLL和任务	(302)
20.1.4 DLL和堆栈	(302)
20.1.5 Windows如何使DLL定位	(303)
20.2 什么时候使用一个定制DLL	(303)
20.2.1 应用程序间的共享	(304)
20.2.2 对于不同市场应用程序的定制	(305)
20.2.3 Windows钩 (hooks)	(305)
20.2.4 设备驱动程序	(306)
20.2.5 通用控制 (custom controls)	(306)
20.2.6 计划管理	(312)
20.3 DLL的建立	(312)
20.3.1 C语言源文件的建立	(313)
20.3.2 模块定义文件的建立	(317)
20.3.3 Make文件的建立	(318)
20.4 应用程序对DLL代码的访问	(320)
20.4.1 库函数原型的建立	(321)
20.4.2 库函数的输入	(321)
20.5 关于Windows对象所有权的规则	(323)
20.6 一个库实例: Select	(324)
20.6.1 建立函数	(324)
20.6.2 建立初始化例程	(328)
20.6.3 建立Exit例程	(329)
20.6.4 建立模块定义文件	(329)
20.6.5 建立包含文件	(329)
20.6.6 编译和连接	(330)
20.7 小结	(330)
第二十一章 多文件界面	(331)
21.1 MDI应用程序的结构	(331)
21.2 初始化MDI应用程序	(332)
21.2.1 寄存窗口类别	(332)
21.2.2 创建窗口	(333)
21.3 编写主消息循环	(333)

21.4	编写边框窗口函数	(334)
21.5	编写子窗口函数	(334)
21.6	将数据与子窗的对应	(335)
21.6.1	将数据存贮于窗口结构中	(335)
21.6.2	使用窗口特性 (Property)	(335)
21.7	控制子窗	(335)
21.7.1	创建子窗	(336)
21.7.2	破坏子窗	(336)
21.7.3	激活和停止子窗	(336)
21.7.4	在屏幕上安排子窗	(337)
21.8	小结	(337)
第二十二章	动态数据交换	(338)
22.1	Windows中的数据交换	(338)
22.1.1	书写板传送	(338)
22.1.2	动态链接库	(338)
22.1.3	动态数据交换	(339)
22.1.4	利用Windows DDE.....	(339)
22.1.5	从用户观点来看DDE	(339)
22.2	DDE概念	(340)
22.2.1	用户程序、服务程序和会话	(340)
22.2.2	应用程序、主题和项	(340)
22.2.3	永久 (“热” 或 “温暖”) 数据链接	(341)
22.3	DDE消息	(341)
22.4	DDE信息流	(341)
22.4.1	初始化一次会话	(342)
22.4.2	传递一个单项	(344)
22.4.3	建立一永久数据链接	(347)
22.4.4	在一个远程应用程序中执行命令	(351)
22.4.5	终止一个会话	(354)
22.5	DDE用户和服务应用程序示例	(355)
22.6	小结	(356)

绪 论

在使用手册之前，应首先阅读本绪论提供的背景知识。

在绪论中讨论下列几个问题：

■在开始前应了解的知识

■本指南的目的和内容

■创建Windows应用程序所必需的工具

■使用本指南讲述的应用程序实例

■本手册中所用符号的约定

■Microsoft Windows™软件开发包（SDK）配带的手册

在开始前首先应了解什么？

在开始使用本指南之前，你应具备：

■使用Windows的经验和对Windows用户界面的了解。

在开始开发Windows应用软件之前，应先在计算机上安装Windows版本3.0，然后学会使用。一定要掌握Windows应用程序（如窗口、会话框、菜单、控制和滚动杆的各部分的名称、目的和操作。因为你自己的Windows应用程序中将含有这些特点所以必须掌握它们，以便正确地实现它们。

■理解Windows用户界面设计风格。

Microsoft Windows的一个目标就是为所有应用程序提供一个共享的用户界面，以此帮助用户减少学习Windows应用程序用户界面带来的麻烦，也可帮助你在设计用户界面时弄清怎样做出选择。为达到这一目标，你必须将你的应用程序用户界面设计基于《系统应用结构、共享用户存取：高级界面设计指南》的用户程序风格进行设计。

■设计C语言程序的经验和使用标准C运行库函数的经验。

C程序设计语言是Windows应用程序开发的最佳语言。Windows的许多特点是为C程序员而设计的。（Windows应用程序也可以用Pascal和汇编语言开发，不过这些语言额外带来了许多麻烦，而用C语言设计应用程序可以避开这些麻烦。）

本指南的内容

本指南的目的是帮助有经验的C程序员利用Microsoft Windows 3.0的应用程序接口设计应用程序。它讲述了如何利用Windows的函数、消息和数据结构来进行高效的编程，并用丰富的程序实例解释了这一过程。这些实例经编译后可在Windows 3.0下运行。

本指南由三部分组成，每部分含若干章。

第一部分是“如何写Windows应用程序”，这里概述了Windows环境并对Windows应用程序实例进行了深入剖析。本部分含下列各章：

■第一章是“Windows环境概述”，这里将Windows环境与C环境进行了一番比较，简单介绍了Windows，描述了Windows程序设计模型和应用程序开发过程。

■第二章是“一个典型的应用程序Generic”，这里示范了如何创建一个简单的Wind-