

 高等教育规划教材

# 计算机图形学

陆 玲 李丽华 宋文琳 桂 颖 编著



提供电子教案

下载网址 <http://www.cmpedu.com>



机械工业出版社  
CHINA MACHINE PRESS



高等教育规划教材

# 计算机图形学

陆 玲 李丽华 宋文琳 桂 颖 编著



机械工业出版社

本书介绍计算机图形学的基本原理及常用图形算法，主要内容包括：Visual C++ 6.0 简介、图形系统、二维图形生成算法、图形变换、图形裁剪、曲线与曲面的生成、消除隐藏线和隐藏面、真实感图形技术、分形图形的生成，以及三维植物造型应用实例等，其中三维植物造型应用实例包含作者的部分科研成果。书中附有常用图形算法的 VC++ 源程序代码。每章均配有习题，可指导读者深入地进行学习，附录为实验指导。

本书可作为计算机及相关专业本科生的教材，也可以作为研究生的参考书或上机指导书，还适用于计算机图形学的初学者。

本书配套授课电子课件，需要的教师可登录 [www.cmpedu.com](http://www.cmpedu.com) 免费注册，审核通过后下载，或联系编辑索取。QQ: 2850823885。电话: 010-88379739。

## 图书在版编目 (CIP) 数据

计算机图形学 / 陆玲等编著. —北京: 机械工业出版社, 2017.3

高等教育规划教材

ISBN 978-7-111-56431-7

I. ①计… II. ①陆… III. ①计算机图形学—高等学校—教材  
IV. ①TP391.41

中国版本图书馆 CIP 数据核字 (2017) 第 063500 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 郝建伟 责任校对: 张艳霞

责任印制: 李 飞

北京机工印刷厂印刷 (三河市南杨庄国丰装订厂装订)

2017 年 4 月第 1 版·第 1 次印刷

184mm×260mm·16.5 印张·399 千字

0001—3000 册

标准书号: ISBN 978-7-111-56431-7

定价: 45.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

网络服务

服务咨询热线: (010) 88379833

机工官网: [www.cmpbook.com](http://www.cmpbook.com)

读者购书热线: (010) 88379649

机工官博: [weibo.com/cmp1952](http://weibo.com/cmp1952)

教育服务网: [www.cmpedu.com](http://www.cmpedu.com)

封面无防伪标均为盗版

金书网: [www.golden-book.com](http://www.golden-book.com)

# 出版说明

当前,我国正处在加快转变经济发展方式、推动产业转型升级的关键时期。为产业转型升级提供高层次人才,是高等院校最重要的历史使命和战略任务之一。高等教育要培养基础性、学术型人才,但更重要的是加大力度培养多层次、多样化的应用型、复合型人才。

为顺应高等教育迅猛发展的趋势,配合高等院校的教学改革,满足各院校对高质量教材的迫切需求,机械工业出版社邀请了全国多所高等院校的专家、一线教师及教务部门,通过充分的调研和讨论,针对相关课程的特点,总结教学中的实践经验,组织出版了这套“高等教育规划教材”。

本套教材具有以下特点:

1) 符合高等院校的人才培养目标及课程体系设置,注重培养学生的应用能力,加大案例篇幅或实训内容,强调知识、能力与素质的综合训练。

2) 针对多数学生的学习特点,采用通俗易懂的方法讲解知识,逻辑性强、层次分明、叙述准确而精炼、图文并茂,使学生可以快速掌握,学以致用。

3) 凝结一线骨干教师的课程改革和教学研究成果,融合先进的教学理念,在教学内容和方法上做出创新。

4) 为了体现建设“立体化”精品教材的宗旨,本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、源代码或源程序、教学大纲、课程设计和毕业设计指导等资源。

5) 注重教材的实用性、通用性,适合各类高等院校、高等职业学校及相关院校的教学,也可作为各类培训班教材和自学用书。

欢迎教育界的专家和老提出宝贵的意见和建议。衷心感谢广大教育工作者和读者的支持与帮助!

机械工业出版社

# 前 言

计算机图形学主要研究计算机表示、处理和绘制图形的原理及算法，是人与计算机之间直观交互的高效手段。随着计算机的发展与应用，计算机图形学已渗透到各个领域，特别是在动画、游戏、可视化和虚拟现实等方面应用较广，是计算机应用的一个主要研究方向。

全国各大院校的计算机专业大都开设了“计算机图形学”这门课程，以满足时代的需求。“计算机图形学”的教材在国内外也较多，而且教材的内容也日益充实，逐渐从纯学术性、侧重于理论推导和分析，发展到增加了许多算法分析、编程指导及程序代码。本课程建议授课学时为 40 小时，实验学时为 20 小时，并要求先修 C 语言。本书中所介绍的程序都是在 Visual C++ 6.0 环境下调试运行通过的。

全书共分 10 章，内容包括 Visual C++ 6.0 简介（第 1 章）、图形系统（第 2 章）、二维图形生成算法（第 3 章）、图形变换（第 4 章）、图形裁剪（第 5 章）、曲线与曲面的生成（第 6 章）、消除隐藏线和隐藏面（第 7 章）、真实感图形技术（第 8 章）、分形图形的生成（第 9 章）和三维植物造型应用实例（第 10 章）。

本书力求做到以下几点。

- 1) 重点介绍计算机图形学中各类基本图形的生成算法及程序设计，使读者学完本教材后能编程实现基本的二维图形到三维真实感图形。
- 2) 详细介绍三维真实感图形生成的全部过程及程序设计。
- 3) 结合作者的科研成果，将科研转化为教学内容，主要体现在第 10 章。
- 4) 强调理论与实践相结合，动脑与动手相结合，附录中给出了实验指导。

本书由陆玲、李丽华、宋文琳、桂颖编著，得到东华理工大学重点教材资助。在此感谢学校的领导和老师给予的大力支持和帮助。受水平所限，书中的不足之处在所难免，恳请广大读者和专家提出宝贵意见。

# 目 录

## 出版说明

## 前言

## 第1章 Visual C++ 6.0 简介 ..... 1

### 1.1 Visual C++开发环境窗口 ..... 1

#### 1.1.1 进入和退出 Visual C++集成开发环境 ..... 1

#### 1.1.2 创建单文档应用程序 ..... 1

#### 1.1.3 添加简单程序代码 ..... 3

### 1.2 Visual C++工程及工作区 ..... 4

#### 1.2.1 工程 ..... 4

#### 1.2.2 工作区 ..... 4

#### 1.2.3 关键类简介 ..... 5

#### 1.2.4 图形设备简介 ..... 6

### 1.3 Visual C++简单程序设计 ..... 7

#### 1.3.1 菜单的设计 ..... 7

#### 1.3.2 对话框的设计 ..... 9

### 习题 1 ..... 12

## 第2章 图形系统 ..... 13

### 2.1 计算机图形学的发展及应用 ..... 13

#### 2.1.1 计算机图形学的发展简史 ..... 13

#### 2.1.2 计算机图形学在我国的发展 ..... 14

#### 2.1.3 计算机图形学的应用 ..... 15

#### 2.1.4 计算机图形学的发展动向 ..... 17

### 2.2 图形生成硬件设备 ..... 18

#### 2.2.1 图形输入设备 ..... 18

#### 2.2.2 图形显示设备 ..... 22

#### 2.2.3 硬拷贝输出设备 ..... 29

### 2.3 图形软件系统 ..... 32

#### 2.3.1 图形软件的组成 ..... 32

#### 2.3.2 基本图形软件 ..... 32

### 习题 2 ..... 33

## 第3章 二维图形生成算法 ..... 34

### 3.1 直线图形 ..... 34

#### 3.1.1 数值微分算法 ..... 34

#### 3.1.2 中点画线算法 ..... 36

#### 3.1.3 Bresenham 画线算法 ..... 38

#### 3.1.4 直线条宽的处理 ..... 40

### 3.2 圆与椭圆图形 ..... 41

#### 3.2.1 简单方程产生圆弧 ..... 41

#### 3.2.2 中点画圆算法 ..... 42

#### 3.2.3 Bresenham 画圆算法 ..... 44

#### 3.2.4 椭圆算法 ..... 45

### 3.3 字符 ..... 47

#### 3.3.1 点阵字符 ..... 47

#### 3.3.2 向量字符 ..... 48

### 3.4 区域填充 ..... 51

#### 3.4.1 种子填充算法 ..... 52

#### 3.4.2 多边形域填充 ..... 58

#### 3.4.3 区域填充图案 ..... 65

### 3.5 图形反走样基础 ..... 70

#### 3.5.1 过取样 ..... 70

#### 3.5.2 简单区域取样 ..... 71

### 习题 3 ..... 71

## 第4章 图形变换 ..... 73

### 4.1 窗口视图变换 ..... 73

#### 4.1.1 窗口区和视图区 ..... 73

#### 4.1.2 窗口区和视图区的坐标变换 ..... 74

### 4.2 图形的几何变换 ..... 75

#### 4.2.1 二维图形的几何变换 ..... 75

#### 4.2.2 三维图形的几何变换 ..... 80

### 4.3 形体的投影变换 ..... 84

#### 4.3.1 投影变换的分类 ..... 84

#### 4.3.2 平行投影 ..... 85

#### 4.3.3 透视投影 ..... 91

#### 4.3.4 投影空间 ..... 98

### 习题 4 ..... 99

第5章 图形裁剪	101	习题6	148
5.1 二维裁剪	101	第7章 消除隐藏线和隐藏面	149
5.1.1 Cohen-Sutherland 直线裁剪算法	101	7.1 隐藏线和隐藏面	149
5.1.2 中点分割算法	105	7.2 Roberts 算法消除隐藏线	149
5.1.3 凸多边形窗口的 Cyrus-Beck 线 裁剪算法	106	7.3 消除隐藏面	154
5.1.4 内裁剪与外裁剪	108	7.3.1 Z 缓冲器算法	154
5.1.5 凹凸多边形的判定	108	7.3.2 画家算法	157
5.1.6 凹多边形的分割算法	109	7.3.3 扫描线算法	158
5.1.7 Sutherland-Hodgman 多边形 裁剪算法	110	7.3.4 可见面光线追踪算法	159
5.1.8 Weiler-Atherton 多边形裁剪算法	114	习题7	161
5.1.9 字符裁剪	115	第8章 真实感图形技术	162
5.2 三维裁剪	116	8.1 颜色	162
5.2.1 三维 Cohen-Sutherland 端点 编码算法	116	8.1.1 色度与三刺激理论	162
5.2.2 三维中点分割算法	118	8.1.2 CIE 色度图	163
5.2.3 三维 Cyrus-Beck 算法	119	8.1.3 颜色系统之间的转换	165
习题5	120	8.1.4 颜色模型	166
第6章 曲线与曲面的生成	121	8.2 简单光照模型	169
6.1 曲线的生成	121	8.3 多边形表示的明暗处理	175
6.1.1 常见参数方程曲线	121	8.3.1 恒定光强的多边形绘制	175
6.1.2 Bezier 曲线	122	8.3.2 Gouraud 明暗处理	175
6.1.3 B 样条曲线	127	8.3.3 Phong 明暗处理	177
6.2 曲面的生成	131	8.4 纹理表示	177
6.2.1 旋转曲面参数方程	132	8.4.1 颜色纹理显示	177
6.2.2 球面	132	8.4.2 凹凸纹理表示	180
6.2.3 圆环面	135	8.5 透明处理与阴影显示	182
6.2.4 圆柱面、圆锥面和圆台面	135	8.5.1 透明处理	182
6.2.5 任意曲线旋转面	137	8.5.2 阴影显示	183
6.3 双线性曲面生成	138	8.6 整体光照模型与光线跟踪算法	186
6.3.1 平面	138	8.6.1 整体光照模型	186
6.3.2 双线性曲面	139	8.6.2 Whitted 整体光照模型	186
6.4 单线性曲面生成	139	8.6.3 光线跟踪算法	187
6.4.1 柱面	139	习题8	188
6.4.2 直纹面	140	第9章 分形图形的生成	189
6.5 Bezier 曲面及其拼合	141	9.1 函数递归分形图形	189
6.5.1 Bezier 曲面	141	9.1.1 Koch 曲线	189
6.5.2 Bezier 曲面的拼合	144	9.1.2 生成元分形图形	190
6.6 B 样条曲面	146	9.1.3 树枝的生成	192
		9.2 L 系统	197
		9.2.1 二维 L 系统	197
		9.2.2 三维 L 系统	202

9.2.3 真实感三维 L 系统 .....	205	10.6.1 百合花开花造型 .....	245
9.3 迭代函数系统 .....	208	10.6.2 南瓜生长造型 .....	246
习题 9 .....	209	10.6.3 荔枝生长造型 .....	247
<b>第 10 章 三维植物造型应用实例</b> .....	<b>210</b>	10.6.4 黄瓜生长造型 .....	248
10.1 参数曲面变形 .....	210	10.7 果实体模造型实例 .....	249
10.2 植物枝干造型 .....	211	10.7.1 西瓜体模造型 .....	249
10.3 植物树叶造型 .....	213	10.7.2 冬枣体模造型 .....	249
10.3.1 植物叶片造型 .....	213	10.7.3 杏体模造型 .....	250
10.3.2 植物叶脉造型 .....	219	习题 10 .....	250
10.3.3 叶片整体造型实例 .....	221	<b>附录 实验指导</b> .....	<b>251</b>
10.4 植物花朵造型 .....	225	实验 1 直线与圆的绘制 .....	251
10.4.1 花瓣的边界造型 .....	225	实验 2 字符绘制 .....	251
10.4.2 花瓣的弯曲造型 .....	226	实验 3 区域填充 .....	251
10.4.3 花瓣颜色模拟 .....	226	实验 4 图形投影变换 .....	252
10.4.4 花冠的造型 .....	229	实验 5 图形裁剪 .....	252
10.4.5 花朵造型实例 .....	229	实验 6 曲线的绘制 .....	252
10.5 植物果实造型 .....	238	实验 7 曲面的绘制 .....	253
10.5.1 椭球及其变形参数方程 .....	238	实验 8 消隐处理 .....	253
10.5.2 果实形状的绘制 .....	239	实验 9 光照模型 .....	253
10.5.3 果实颜色纹理模拟 .....	240	实验 10 综合处理 .....	254
10.5.4 果实造型实例 .....	241	<b>参考文献</b> .....	<b>255</b>
10.6 植物生长造型 .....	245		

# 第 1 章 Visual C++ 6.0 简介

Visual C++是微软公司开发的一个IDE（集成开发环境），是一个功能强大的可视化软件开发工具。Visual C++应用程序的开发主要有两种模式，一种是 WIN API 方式，另一种是 MFC 方式。传统的 WIN API 方式比较烦琐，而 MFC 则是对 WIN API 的再次封装。MFC（Microsoft Foundation Class）库是一整套简化 Windows 编程的可重用的类库，提供了 Windows 编程常用类。MFC 库应用程序框架包含自己的应用程序结构，使用 MFC 库编写 Windows 程序，有利于代码的维护和增强。使用 MFC 类库，程序可以在任何时候调用 Win32 函数，可以最大程度地利用 Windows。本章介绍了使用 Visual C++开发图形应用程序的一些基本技术。

## 1.1 Visual C++开发环境窗口

### 1.1.1 进入和退出 Visual C++集成开发环境

启动并进入 Visual C++ 6.0 集成开发环境通常有 3 种方法。

- 1) 单击“开始”按钮，选择“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0 命令。
  - 2) 在桌面上创建 Microsoft Visual C++ 6.0 的快捷方式，直接双击该图标。
  - 3) 如果已经创建了某个 Visual C++工程，双击该工程的 dsw（Develop Studio Workshop）文件，也可以进入集成开发环境，并打开该工程。
- 选择“文件”→“退出”命令，可退出集成开发环境。

### 1.1.2 创建单文档应用程序

下面以单文档应用程序为例，说明如何创建一个简单的应用程序。

- 1) 进入 Visual C++ 6.0 集成开发环境后，选择“文件”→“新建”命令，弹出“新建”对话框。选择“工程”选项卡，在其左边的列表框中选择 MFC AppWizard (exe) 工程类型，在“工程名称”文本框输入工程名，在“位置”中选择工程路径（如选择 D 盘），则将在 D 盘下建立一个新的以工程名命名的目录。如果是第一个工程文件，则必须创建一个新的工作区，选择“创建新的工作空间”单选按钮，在“平台”列表框中选择 Win32，如图 1-1 所示。

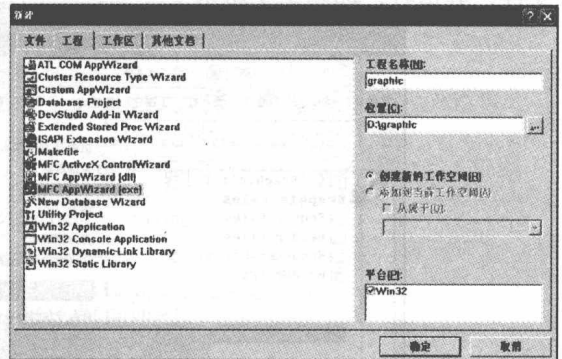


图 1-1 “新建”对话框

- 2) 单击“确定”按钮，弹出“MFC 应用程序向导-步骤 1”对话框，选择“单文档”单选按钮，如图 1-2 所示。
- 3) 单击“完成”按钮，弹出如图 1-3 所示的“新建工程信息”对话框。

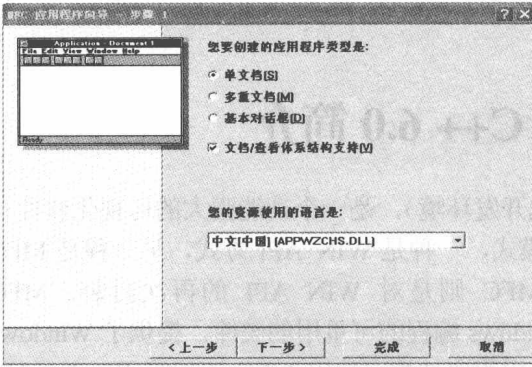


图 1-2 “MFC 应用程序向导-步骤 1”对话框

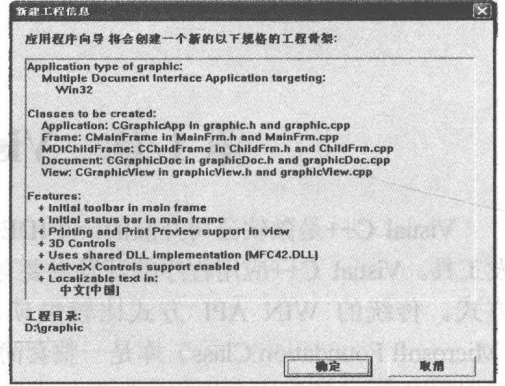


图 1-3 “新建工程信息”对话框

4) 单击“确定”按钮，完成应用程序的自动生成，在指定的目录下生成了应用程序框架所必需的全部文件，并且可以直接运行，如图 1-4 所示。

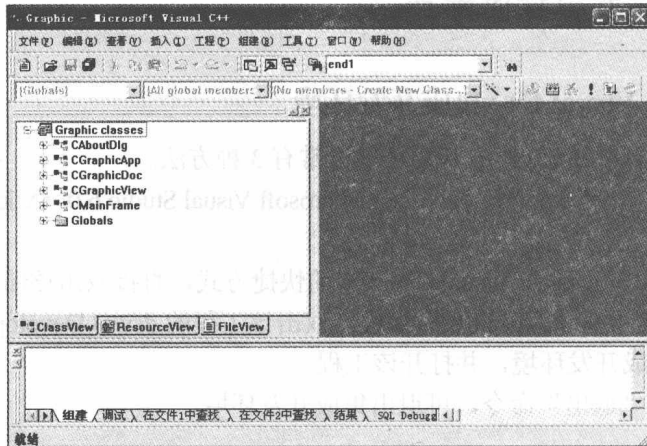


图 1-4 应用程序界面

5) 选择“组建”→“执行”命令，如图 1-5 所示。因为是第一次执行，没有生成可执行文件.exe，提示是否生成（见图 1-6），单击“是”按钮，则系统进行编译及链接，生成可执行文件，并运行，如图 1-7 所示。

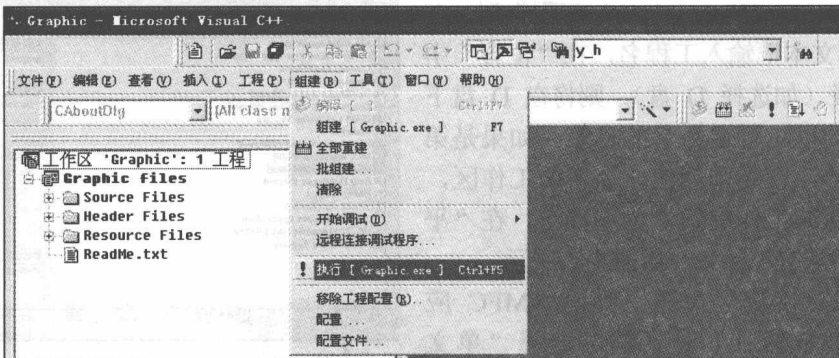


图 1-5 执行程序

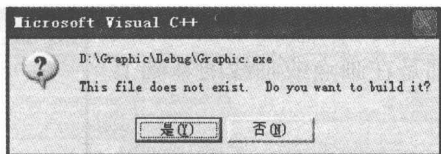


图 1-6 提示是否生成执行程序

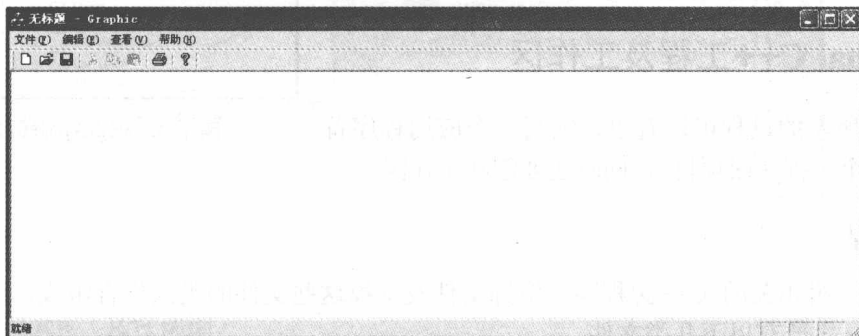


图 1-7 程序运行界面

用户可在此程序框架下添加自己的程序代码。

在 Visual C++ 中，代码主要有 4 种颜色：黑、蓝、绿和灰。黑色是普通代码的标志；蓝色标识关键字，包括 if、for 等程序流程关键字和 int、float 等数据类型关键字，但用 typedef 或 #define 生成的新类型不被标识；绿色标识程序的注释，即在 /\*...\*/ 之间的部分或以 // 开头的行；灰色的代码表示是由 Class Wizard 自动生成的代码，一般情况下不要修改。

### 1.1.3 添加简单程序代码

在窗口左边工作区的 FileView 选项卡中，选择 graphicView.cpp 文件，在 void CGraphicView::OnDraw(CDC\* pDC) 函数中添加如下代码（见图 1-8）。

```
pDC->SetPixel(100,100,RGB(0,0,0));    //在 (100,100) 坐标处绘制一个黑色点
pDC->MoveTo(50,200);                 //将画笔移到 (50,200) 处
pDC->LineTo(100,220);                 //从当前位置、用当前画笔画线到 (100, 220)
```

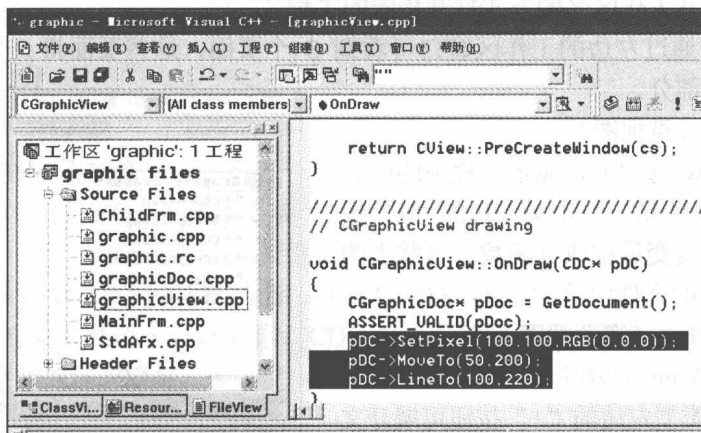


图 1-8 添加画点画线函数

运行程序，得到如图 1-9 所示的画点及画线结果。  
计算机图形学中的绘图算法都是在画点或画线的基础上进行的。

窗口的坐标原点一般默认在左上角。

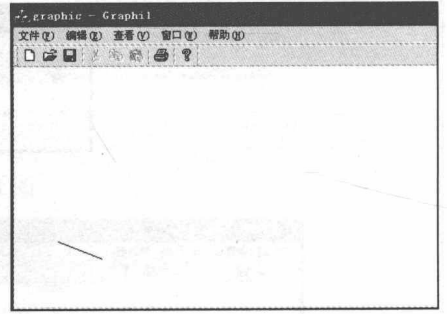


图 1-9 画点与画线的结果

## 1.2 Visual C++工程及工作区

从前面所示的过程可以看出，编写一个应用程序首先要创建一个工程（或项目），同时还要创建工作区。

### 1.2.1 工程

工程由一组相关的 C++源程序、资源文件及支撑这些文件的类的集合组成，全部在工程名的目录下，主要有以下几类文件。

\*.dsp (Developer Studio Project): 工程配置文件，如工程包含哪个文件，编译选项是什么等，编译时是按照.dsp 的配置进行的。

\*.dsw (Developer Studio Workspace): 工作区文件，用来配置工程文件。它可以指向一个或多个.dsp 文件。

\*.clw: ClassWizard 信息文件。

\*.opt: 工程关于开发环境的参数文件，如工具条位置等信息。

\*.rc: 资源文件。位图或菜单之类的资源。

\*.plg: 编译信息文件，编译时的 error 和 warning 信息文件。

\*.ncb (No Compile Browser): 无编译浏览文件。

\*.cpp: 源代码文件，按 C++语法编译处理。

\*.h: 头文件，一般用作声明和全局定义。

### 1.2.2 工作区

工作区用工作区文件.dsw 来描述，工作区文件保存了集成开发环境中应用程序的工程设置信息，一般用打开工作区名的方式打开指定的工程。

创建工程后，通过左边的工作区窗口可看到 3 个页面，这些页面将一个工程按照一定的逻辑关系分成几个部分。

#### 1. ClassView (类视图)

选择 ClassView 选项卡，显示工程中的所有 C++类，如图 1-10 所示。单击类左边的“+”按钮可列出该类的成员变量和成员函数。这些类的定义都在 FileView 的文件中。

#### 2. ResourceView (资源视图)

选择 ResourceView 选项卡，列出工程中的所有资源。单击资源类型左边的“+”按钮可展开文件夹。双击其中的资源可以打开对应的资源编辑

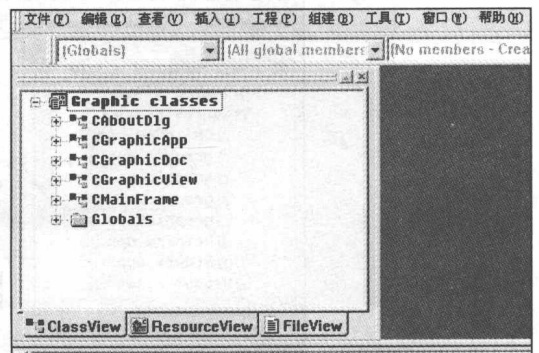


图 1-10 类视图

器，可以对资源进行编辑，如图 1-11 所示。

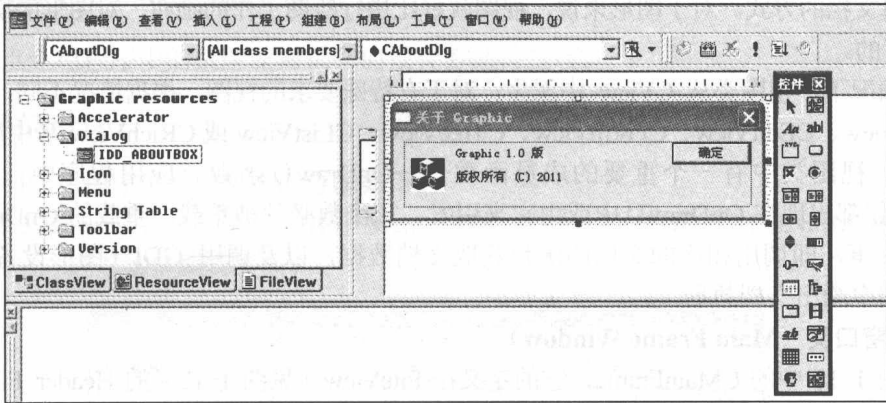


图 1-11 资源视图

### 3. FileView (文件视图)

选择 FileView 选项卡，列出工程中的所有文件及隶属关系。单击文件类型左边的“+”按钮可列出该类型的所有文件。双击其中的一个文件即可打开该文件，可对该文件进行编辑，如图 1-12 所示。

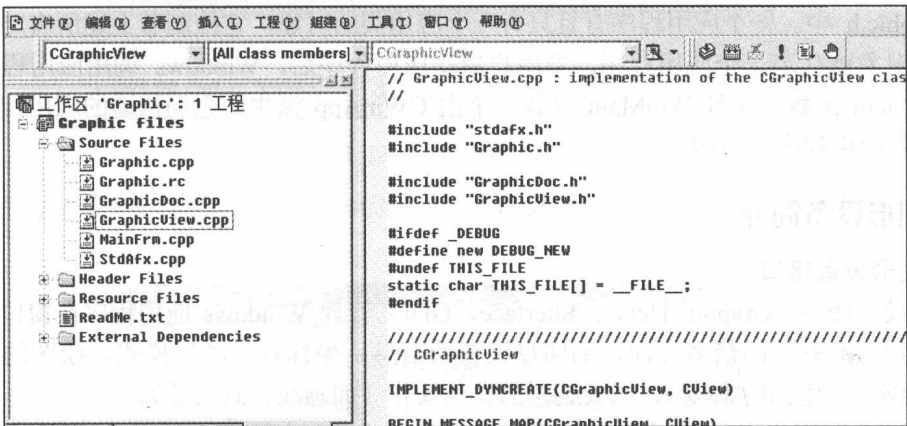


图 1-12 文件视图

## 1.2.3 关键类简介

本节介绍几个关键类。

### 1. 文档类 (Document)

对应图 1-10 中的 CGraphicDoc，其文件名中间的 Graphic 是工程文件名，它的定义在 FileView (见图 1-12) 的 Header Files 类型文件的 GraphicDoc.h 中。文档类一般从 MFC 中的类 CDocument 中派生；如果支持 OLE 功能，可从 ColeDocument 或 ColeServerDoc 类中派生。由 CDocument 派生的类主要用于存储数据。CDocument 类用于相应数据文件的读取，以及存储 CView 类所需观察和处理的信息。

### 2. 视图类 (View)

对应图 1-10 中的 CGraphicView，它的定义在 FileView (见图 1-12) 的 Header Files 类型文

件的 `GraphicView.h` 中。视图相当于文档在应用程序中的观察窗口，确定了用户对文档的观察方式和用户编辑文档的方式。对于图形来说，视图就好比进行绘图工作的画布，对图形的操作都是在视图上进行的。

一般情况下，视图类从 `CView` 中派生；对于有特殊要求的视图，根据情况不同，还可以从类 `CScrollView`、`CEditView`、`CFormView`、`CTreeView`、`CListView` 或 `CRichView` 等中派生。

另外，视图类中有一个重要的成员函数——`OnDraw()` 函数。应用程序中，几乎所有“画”的动作都出现在 `OnDraw()` 中或由它来引发。该函数必须被重载。重载的 `OnDraw()` 函数要完成两件事，即调用相应的文档的函数获取文档数据，以及调用 GDI（图形设备接口）的函数在视图中画出文档数据。

### 3. 主窗口类 (Main Frame Window)

对应图 1-10 中的 `CMainFrame`，它的定义在 `FileView`（见图 1-12）的 Header Files 类型文件的 `MainFrm.h` 中。主窗口是 Windows 应用程序中限定其所有窗口范围的最外边框。应用程序中的所用其他窗口都直接或间接地是主窗口的子窗口，如标准菜单、工具条和状态条等。一个应用程序一般具有主窗口类。SDI 应用程序的主窗口类应从类 `CFrameWnd` 中派生，MIDI 程序的主窗口类应从类 `CMDIFrameWnd` 中派生。

### 4. 应用类 (Application)

对应图 1-10 中的 `CGraphicApp`，它的定义在 `FileView`（见图 1-12）的 Header Files 类型文件的 `Graphic.h` 中。一个应用程序有且只有一个应用类的对象，它控制着上述所有对象。一个应用程序对象就代表一个应用程序，当用户启动应用程序后，Windows 调用应用程序框架内置的 `WinMain` 函数，并且 `WinMain` 寻找一个由 `CWinApp` 派生的全局构造的应用程序对象，全局对象在应用程序之前构造。

## 1.2.4 图形设备简介

### 1. 图形设备接口

图形设备接口 (Graphic Device Interface, GDI) 管理 Windows 应用程序在窗口中的所有绘图操作和与此有关的诸多方面，如图形设备的信息、坐标系和映射模式、绘图的当前状态（画笔、画刷、颜色和字体等），以及绘图的具体操作（如画线、画圆等）。

Windows 图形设备接口对象类型由 MFC 类库表示，这些类有一个共同的抽象基类：`CGdiObject`。Windows 图形设备接口对象由 `CGdiObject` 派生类的 C++ 对象来表示，这些对象有以下几个。

- `CBitmap`: 位图对象。
- `CBrush`: 画刷。用于表示区域填充的颜色和样式。
- `CPen`: 画笔。用于指定线和边框的性质，如颜色、线宽和线形等。
- `CFont`: 字体。具有一定大小和风格的一套字符集。

### 2. 设备环境类

`CDC` 是 MFC 中最重要的类之一，更是绘图应用程序中最重要的类。`CDC` 提供的成员函数可以用于对设备环境的操作、绘图工具的使用和图形设备接口 (GDI) 对象的选择等。在使用 `CDC` 对象时，必须先构造一个 `CDC` 对象，然后才能调用它的成员函数。使用完成后，必须在适当的地方将其删除，在 Windows 环境中可获得的设备环境的数量是有限的。如果太多的 `CDC` 对象没有被删除，计算机的资源将很快被耗尽，VC++ 也会在调试窗口中报错。

## 1.3 Visual C++简单程序设计

本节重点介绍程序中的菜单及对话框的设计方法与过程。

### 1.3.1 菜单的设计

在单文档应用程序中设计菜单的方法如下。

1) 选择 ResourceView 选项卡, 展开 Menu 选项, 双击其下的 IDR\_MAINFRAME 文件, 如图 1-13 所示, 进入菜单编辑界面。

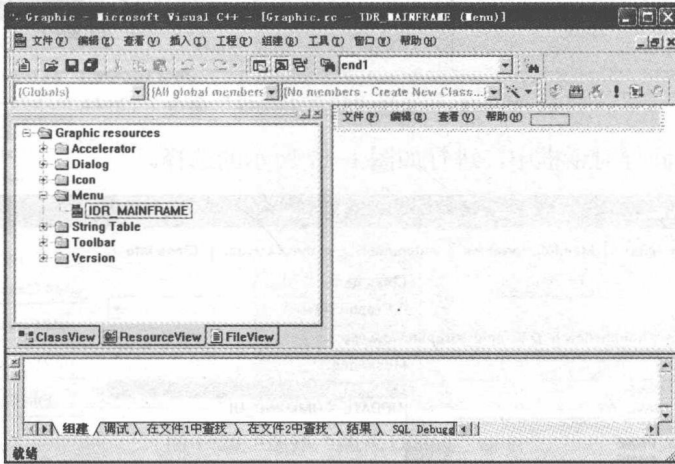


图 1-13 菜单编辑界面

2) 在右边菜单空栏上右击, 在弹出的快捷菜单中选择“属性”命令, 弹出如图 1-14 所示的对话框。

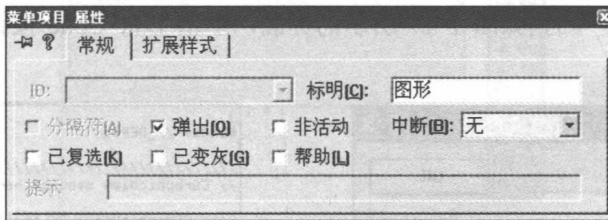


图 1-14 “菜单项目属性”对话框

3) 在“图形”菜单的子菜单空栏处右击, 在弹出的快捷菜单中选择“属性”命令, 弹出如图 1-15 所示的对话框, 设置其子菜单属性。

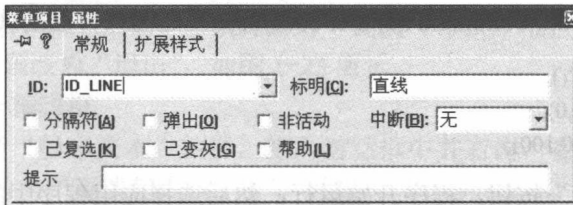


图 1-15 子菜单属性设置

4) 在菜单编辑界面处右击，在弹出的快捷菜单中选择“建立类向导”命令，如图 1-16 所示。

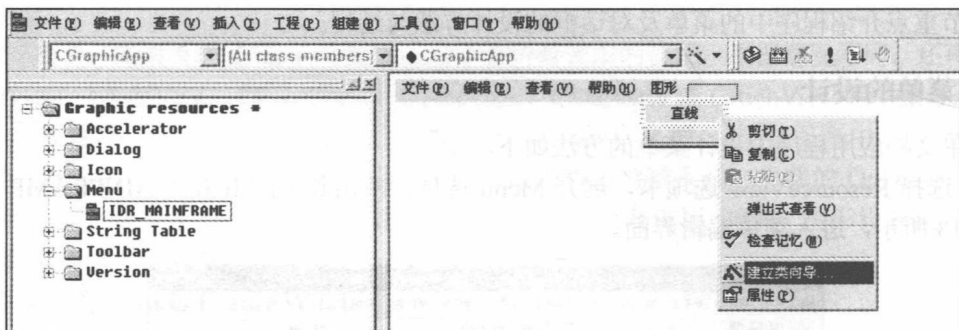


图 1-16 选择“建立类向导”命令

5) 在弹出的类向导对话框中，进行如图 1-17 所示的选择。

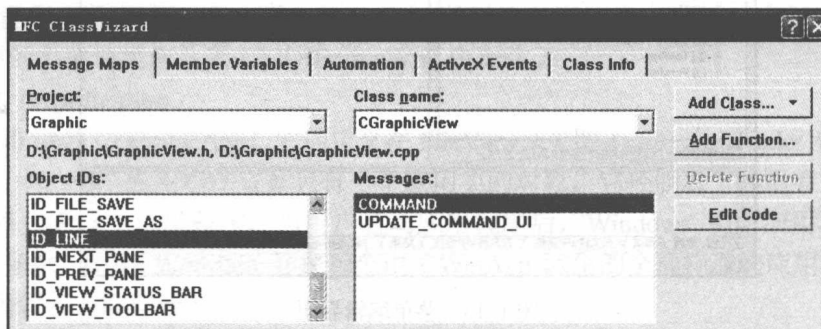


图 1-17 MFC ClassWizard 对话框

6) 单击 Add Function 按钮，弹出如图 1-18 所示的对话框，添加一个成员函数。

7) 单击 OK 按钮，回到如图 1-17 所示的界面，单击 Edit Code 按钮，进入如图 1-19 所示的代码编辑界面。

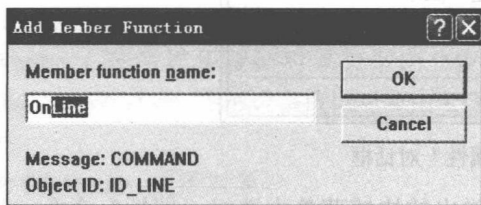


图 1-18 “添加成员函数”对话框

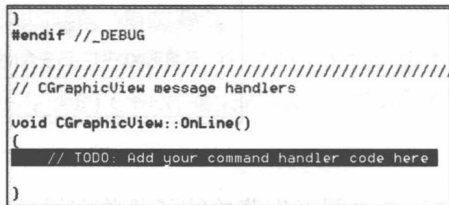


图 1-19 代码编辑界面

在“// TODO: Add your command handler code here”处输入自己的代码，例如：

```
CDC pDC=GetDC();
pDC->MoveTo(0,0);
pDC->LineTo(100,100);
```

单击工具栏中的“!”按钮，程序开始运行，然后选择应用程序中的“图形”→“直线”命令，就可以绘制出一条直线。

## 1.3.2 对话框的设计

如果直线的起点坐标与终点坐标需要用户在对话框中输入，则需要建立对话框，建立对话框的步骤如下。

### 1. 建立新对话框

选择 ResourceView 选项卡，选择 Dialog 选项并右击，在弹出的快捷菜单中选择“插入 Dialog”命令（见图 1-20），进入对话框编辑界面，如图 1-21 所示。

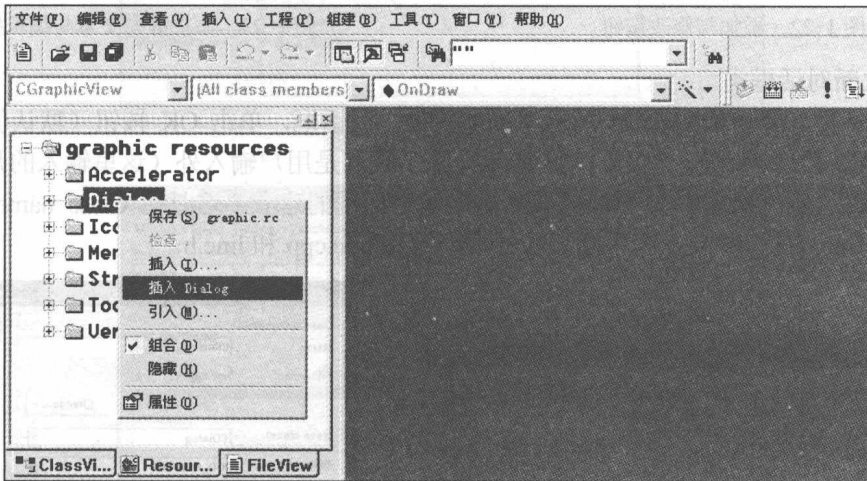


图 1-20 选择“插入 Dialog”命令

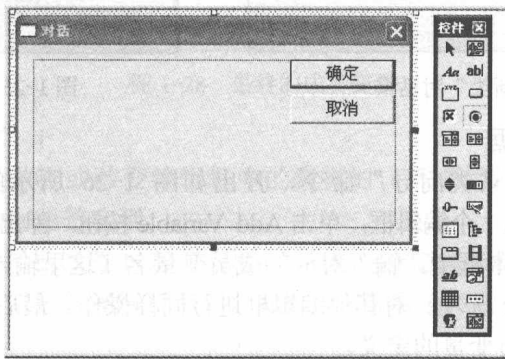


图 1-21 对话框编辑界面

### 2. 添加与修改按钮

在控件工具内选择“按钮”控件，拖入对话框中并右击，在弹出的快捷菜单中选择“属性”命令，修改 ID 值为 IDC\_LINE，修改标题为“直线”。同时，删除已有的“确定”按钮，并将“取消”按钮的标题改为“退出”，如图 1-22 所示。

### 3. 添加静态文本与编辑框

在控件工具内选择“静态文本”控件，拖入对话框中并右击，在弹出的快捷菜单中选择“属性”命令，将标题改为“直线起点(x, y)”。再拖入一个静态文本，设置标题为“直线终点(x, y)”。

在控件工具内选择“编辑框”控件，拖入对话框中，共拖入 4 个编辑框，如图 1-23 所示。