

大学计算机规划教材



Visual Basic.NET 程序设计 (第3版)

◆ 陆汉权 冯晓霞 方红光 编著



电子工业出版社

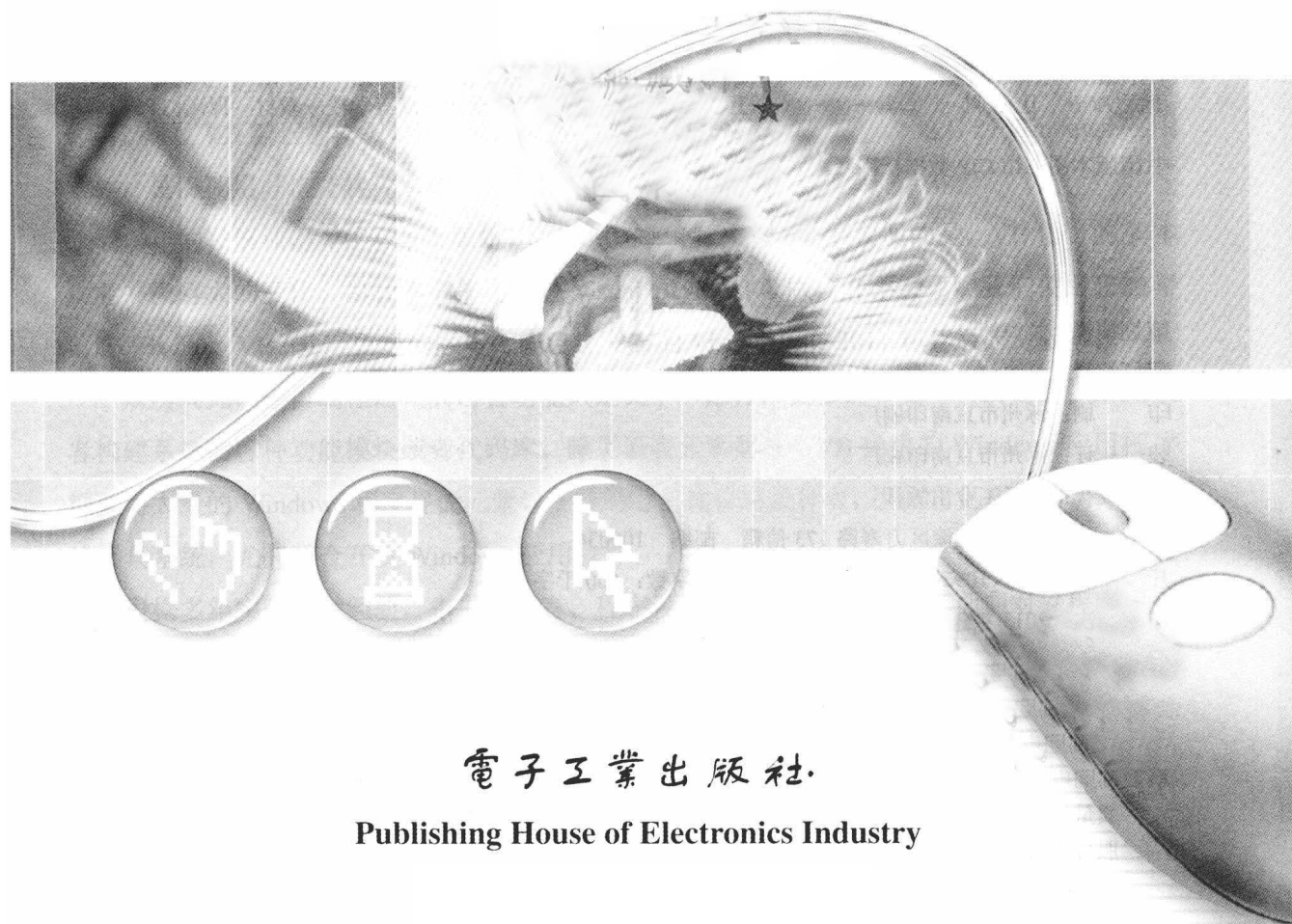
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY <http://www.phei.com.cn>

大学计算机规划第



Visual Basic.NET 程序设计(第3版)

◆ 陆汉权 冯晓霞 方红光 编著



電子工業出版社

Publishing House of Electronics Industry

内 容 简 介

本书是浙江大学计算机基础类课程建设的成果。本书以 Visual Basic 2008 作为开发平台, 共 17 章, 从介绍学习程序设计的目的及了解程序运行环境入手, 讲述了如下内容: 数据类型与表达式, 结构化编程的三种结构, 结合 VB 控件进行编程, 用户界面和用户交互编程, 数组及相关控件编程, 过程和函数编程, 字符串和日期处理, 图形处理, 文件和数据库编程, 网络和多媒体编程等, 以及面向对象程序设计的基本过程。本书在介绍概念和语法的同时提供了大量的例子。

本书既可以作为高等学校程序设计课程的教学用书, 也可作为技术人员的参考用书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目(CIP)数据

Visual Basic .NET 程序设计 / 陆汉权, 冯晓霞, 方红光编著. —3 版. —北京: 电子工业出版社, 2013.3
大学计算机规划教材
ISBN 978-7-121-19827-4

I. ①V… II. ①陆… ②冯… ③方… III. ①BASIC 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 049425 号

策划编辑: 章海涛

责任编辑: 章海涛 特约编辑: 曹剑锋

印 刷: 涿州市京南印刷厂

装 订: 涿州市京南印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 23.5 字数: 600 千字

印 次: 2013 年 3 月第 1 次印刷

定 价: 42.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

大学中的程序设计课程早就成为了大学生的必修课。作为学习者，我们应该理解每门课程的学习目的。对程序设计课程，当然专业学生是作为专业课程学习的，而绝大多数非计算机专业学生学习程序设计，显然不是为了使自已成为程序员 (Programmer)，我们的理解是通过学习程序设计，进一步弄清计算机是如何“按程序设计的步骤去处理、解决问题的”。因此，理解问题并分解问题，使之能适合计算机的处理，是非计算机类专业程序设计课程教学的主要任务。事实上，学习程序设计是进一步理解计算机的最好途径。只有理解了计算机，才可以让计算机能更好地为我们服务。

Visual Basic (简称 VB) 是我国高校计算机基础课程系列中的程序设计基础与实验的主要内容之一。许多人认为它简单易学，当然这只是 VB 的一个特点，它的另一个重要特性是它在以 Windows 为环境的计算机系统中具有特殊地位：VB 作为 Windows 的编程语言不但为计算机编程专业人员所广泛采用，而且由于它延续和继承了 Windows 系统的丰富资源，如窗口、菜单、工具栏及对话框、按钮等常用的交互方法，因此非计算机专业人员能够很快掌握其编程方法，更重要的是，由于 Windows 的实际垄断，通过 VB 理解计算机有着很大的帮助作用。通过 VB 编程的学习，我们就能更容易理解诸如 Microsoft Office 这样的 Windows 应用软件，进而使用好这些软件系统。

多年的发展使得 VB 有多个版本，如 2008 版、2010 版等。要说明的是，VB 语言的基本功能并没有大的变化。尽管本书采用的是 Visual Basic 2008 版，并不会影响我们使用它的早前的 VB 5.0、VB 6.0 或更早的版本，或已经发布的 Visual Basic 2011 和其后的新版本。

众所周知，任何编程语句其语言功能只是其中一部分，更多的和更重要的是提供给编程者构建系统的各种功能模块或者代码库。除了语言元素部分，VB 提供的 Windows 应用程序框架将大量的 Windows 程序界面元素以工具的方式提供给设计者，因此它更是一种软件开发的工具系统，尤其适合开发 Windows 应用软件。

根据多年实际教学经验，本书把语言、程序设计、算法、程序架构、面向对象程序设计等和 VB 特点相关的许多知识点进行了组合，并给出了学习程序设计的目的和提出了学习程序设计是掌握计算机方法的重要途径的观点，既适合初学计算机程序设计的读者，也能够为需要提升 VB 编程能力的读者提供帮助。

全书共 17 章，每章开始先导入基本概念，给出学习目标，并以示例程序作为引导，帮助读者建立对本章相关知识的初步认识，然后展开介绍。这种结构源于我们多年的教学实践，对计算机语言类课程特别是 VB 这类实践性非常强且比较抽象的基础课程，采用这种教学过程是最适合的。本书特别考虑了计算机基础较弱的读者，有针对性地结合基础知识进行介绍。

书中设计了大量生动活泼的例题，基本上覆盖了常用算法，也有习题、思考题和上机实验题。读者结合书中的例题进行上机实践，不但能够进一步理解算法以及设计过程，而且能够迅速掌握编程方法，提高编程技巧。

书中包括了一些教学要求范围之外的编程内容，如网络编程、多媒体编程等，在章节前用“*”做了标记，供读者取舍。我们希望能够给读者一个 VB 程序设计的全貌，可以充分领略 VB 编程的魅力。

本书第 1、2、6、9、15 章由陆汉权编写，第 3、4、5、7、8、10、16 章由冯晓霞编写，第 11、12、13、14、17 章由方红光编写。全书由冯晓霞统稿。

本书在编写过程中得到了浙江大学计算机学院沈钦仙、陈建海、章文等老师的大力支持和帮助。作者在此深表感谢之意。本书在出版过程中得到了电子工业出版社的全力支持和帮助，在此也一并表示谢忱之意！

由于作者的水平有限，书中的错误在所难免，恳切希望使用本书的老师 and 读者指正。

本书为任课教师提供配套的教学资源（包含电子教案），需要者可登录华信教育资源网（<http://www.hxedu.com.cn>），注册之后进行免费下载，或发邮件到 unicode@phei.com.cn 进行咨询。

作 者

于杭州，浙大紫金港

反侵权盗版声明

电子工业出版社依法对本作品享有专有出版权。任何未经权利人书面许可，复制、销售或通过信息网络传播本作品的行为；歪曲、篡改、剽窃本作品的行为，均违反《中华人民共和国著作权法》，其行为人应承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。

为了维护市场秩序，保护权利人的合法权益，我社将依法查处和打击侵权盗版的单位和个人。欢迎社会各界人士积极举报侵权盗版行为，本社将奖励举报有功人员，并保证举报人的信息不被泄露。

举报电话：(010) 88254396；(010) 88258888

传 真：(010) 88254397

E-mail: dbqq@phei.com.cn

通信地址：北京市万寿路 173 信箱

电子工业出版社总编办公室

邮 编：100036

目 录

第 1 章 概述	1	2.5.3 算术运算符的优先规则	32
1.1 为何要学习程序设计	1	2.6 VB 编程的开始: 使用窗体	33
1.1.1 什么是程序设计	1	2.6.1 一个使用窗体显示文本的例子	33
1.1.2 有哪几种常用的高级语言	2	2.6.2 窗体的属性	34
1.1.3 翻译计算机语言程序	2	2.6.3 窗体的方法	35
1.1.4 程序设计是一种方法学	3	2.6.4 窗体识别的动作: 事件	36
1.1.5 如何学习程序设计	3	2.6.5 使用对象浏览器	37
1.2 什么是 VISUAL BASIC.NET	4	2.6.6 Tab 键和焦点	37
1.3 VB.NET 平台: 集成开发环境	5	2.7 赋值和输入/输出	38
1.4 项目管理	9	2.7.1 如何进行赋值操作	38
1.4.1 项目文件和窗体文件	9	2.7.2 使用输入对话框 InputBox 函数	39
1.4.2 使用项目	10	2.7.3 使用消息对话框 MsgBox 函数	40
1.4.3 使用解决方案资源管理器	11	2.8 程序举例	42
1.5 编写一个简单的 VB.NET 窗体		习题 2	44
应用程序	12	实验 2	45
1.6 编写一个简单的 VB.NET 控制台		第 3 章 设计顺序结构程序	46
应用程序	14	3.1 概述	46
1.7 什么是 VB.NET 的面向对象编程	16	3.2 求摄氏温度对应的华氏温度	47
习题 1	17	3.3 Visual Basic 2008 语法和注释	48
实验 1	19	3.4 语句和函数	49
第 2 章 VB.NET 编程入门	21	3.4.1 语句	49
2.1 概述	21	3.4.2 函数	50
2.2 一个简单的加法器程序	22	3.4.3 常用函数举例	50
2.3 确定数据的性质: 数据类型	24	3.5 使用基本控件	54
2.3.1 基本数据类型	25	3.5.1 命令按钮	54
2.3.2 数值类型	25	3.5.2 标签	55
2.3.3 字符数据类型	26	3.5.3 文本框	56
2.3.4 逻辑型与日期型	27	3.5.4 滚动条	60
2.3.5 对象引用型	27	3.5.5 微调器	61
2.3.6 自定义数据类型	27	3.5.6 滑块控件	62
2.4 两种数据表达: 常量和变量	28	3.6 使用计时器控件	63
2.4.1 常量	28	3.7 调试程序	64
2.4.2 符号常量	28	3.8 程序举例	66
2.4.3 变量和声明变量	29	习题 3	68
2.4.4 强制声明变量: Option Explicit	31	实验 3	69
2.4.5 关键字	31	第 4 章 设计选择结构程序	71
2.5 算术运算符和表达式	31	4.1 概述	71
2.5.1 算术运算符	31	4.2 设置文本框字体字型	72
2.5.2 算术表达式	32	4.3 进行比较运算	74

4.4 使用 If 条件语句	75	6.4.1 一般按键: KeyPress 事件	116
4.4.1 使用行 If 语句	75	*6.4.2 特殊按键处理: KeyDown 和 KeyUp	117
4.4.2 使用块 If 语句	76	*6.5 组合: 键盘 Shift、Ctrl、Alt 和鼠标键	118
4.4.3 编写嵌套的选择结构程序	77	6.6 设计菜单	119
4.5 使用 Select Case 条件语句	77	6.6.1 什么是菜单	119
4.6 使用选择控件和分组框	79	6.6.2 使用 MenuStrip 控件	119
4.6.1 单选按钮	79	6.6.3 设计菜单	121
4.6.2 复选框	81	6.6.4 子菜单	121
4.6.3 分组框	82	6.6.5 菜单编程: 菜单的 Click 事件	122
4.7 如何实现程序的转移控制	83	*6.7 快捷方式: 弹出式菜单	123
4.8 程序举例	84	6.8 程序举例	125
习题 4	86	习题 6	127
实验 4	87	实验 6	128
第 5 章 设计循环结构程序	88	第 7 章 使用数组	130
5.1 概述	88	7.1 概述	130
5.2 判断素数	89	7.2 给运动员排名次	130
5.3 进行逻辑运算	90	7.3 一维数组	132
5.4 使用 For/Next 循环语句	92	7.3.1 如何声明一维数组	132
5.4.1 For/Next 语句	92	7.3.2 使用一维数组	133
5.4.2 Exit For 语句	94	7.3.3 使用 For Each/Next 语句访问数组	136
5.5 使用 Do/Loop 循环语句	94	7.4 使用列表框和组合框控件	136
5.5.1 Do/Loop 语句	95	7.4.1 列表框	136
5.5.2 Exit Do 语句	96	7.4.2 复选列表框	140
*5.6 使用 While/End While 循环语句	97	7.4.3 组合框	140
5.7 如何实现多重循环控制	98	7.5 使用一维控件数组	143
5.8 程序举例	99	7.6 使用二维数组	145
5.8.1 辗转相除法	99	7.7 使用可调数组	148
5.8.2 递推法	100	7.8 程序举例	149
5.8.3 穷举法	101	7.8.1 插入问题	150
5.8.4 求水仙花数	102	7.8.2 检索问题	151
5.8.5 验证哥德巴赫猜想	103	7.8.3 选择联系人	152
5.8.6 数制转换	104	习题 7	154
习题 5	105	实验 7	156
实验 5	107	第 8 章 子过程和函数过程	158
第 6 章 用户交互: 使用鼠标、键盘和菜单	109	8.1 概述	158
6.1 概述	109	8.2 查询电话号码	159
6.2 使用鼠标事件的示例程序	110	8.3 窗体模块和标准模块	161
6.3 使用鼠标: 指针、鼠标键和事件	111	8.4 Sub 过程	161
6.3.1 程序状态: 改变鼠标指针的形状	111	8.4.1 定义 Sub 过程	161
6.3.2 确定按下了哪个鼠标键	112	8.4.2 调用 Sub 过程	162
6.3.3 轻松的交互: 鼠标事件	114	8.4.3 Exit Sub 语句	165
*6.3.4 鼠标拖放	114	8.5 Function 函数过程	165
6.4 使用键盘	116	8.6 引用调用和按值调用	168

8.7 存储类和作用域	172	10.4.5 画扇形——DrawPie 方法	216
8.8 程序举例	176	10.4.6 画多边形——DrawPolygon 方法	216
习题 8	180	10.4.7 显示文本——DrawString 方法	217
实验 8	182	10.5 应用绘图笔与图案填充	217
第 9 章 处理字符串和日期时间	183	10.5.1 更改线条形式	217
9.1 概述	183	10.5.2 Brush 对象	219
9.2 一个简单的字符加密处理程序	184	10.5.3 图形填充	221
9.3 字符类型和字符数组	185	10.6 VB 图像	222
9.3.1 字符和字符串	185	10.6.1 图片框	222
9.3.2 比较字符	185	10.6.2 ImageList 控件	225
9.3.3 String 函数和 Str 函数	186	10.6.3 图形复制	226
9.3.4 字符数组	187	10.7 程序举例	228
9.4 操作字符串	189	10.7.1 时钟	228
9.4.1 Len 函数和 Space 函数	189	10.7.2 制作饼图	229
*9.4.2 Like 运算符	190	10.7.3 滚动显示大图片	230
9.4.3 使用 Mid 操作字符串中的字符	190	习题 10	232
9.4.4 使用 Left、Right 和 InStr 函数 操作字符串	191	实验 10	233
9.4.5 删除多余的空格	191	第 11 章 设计用户界面	234
9.4.6 转换函数	191	11.1 概述	234
9.4.7 类型转换函数	192	11.2 使用用户名和密码登录应用程序	235
9.4.8 格式化函数 Format	193	11.3 设计多窗体应用程序	236
*9.5 查找和替换字符串	194	11.4 设计和用户交互的对话框	240
9.6 处理日期和时间	196	11.5 使用通用对话框	241
9.6.1 日期函数和数据类型	196	11.5.1 “打开”对话框	241
9.6.2 时间函数和时间数据	197	11.5.2 “另存为”对话框	243
9.6.3 日期的计算	198	11.5.3 “颜色”对话框	243
9.6.4 日期和时间的格式化	199	11.5.4 “字体”对话框	244
*9.7 使用日历控件 MonthCalendar	201	11.5.5 “浏览文件夹”对话框	245
9.8 程序举例	203	11.5.6 “打印”对话框	245
习题 9	204	11.6 设计工具栏	245
实验 9	205	11.6.1 通过手工方式创建工具栏	245
第 10 章 处理图形	206	11.6.2 使用工具栏控件创建工具栏	247
10.1 概述	206	11.7 使用状态栏	249
10.2 模拟单摆	207	11.8 程序举例	251
10.3 绘图相关对象	208	习题 11	253
10.4 使用绘图方法	212	实验 11	254
10.4.1 画直线——DrawLine 和 DrawLines 方法	212	第 12 章 处理文件	255
10.4.2 画矩形——DrawRectangle 或 DrawRectangles 方法	213	12.1 概述	255
10.4.3 画圆、画椭圆——DrawEllipse 方法	213	12.2 操作顺序文件数据的例子	256
10.4.4 画弧——DrawArc 方法	216	12.3 使用与文件、目录有关的方法	257
		12.4 操作顺序文件	259
		12.4.1 什么是顺序文件	259

12.4.2	打开顺序文件	260	14.6	ASP.NET Web 编程简介	306
12.4.3	写入操作	260	14.6.1	ASP.NET 网页概述	307
12.4.4	关闭顺序文件	262	14.6.2	ASP.NET 服务器控件	307
12.4.5	读出操作	262	14.6.3	创建 Web 应用程序	308
12.5	操作随机文件	264	14.6.4	ASP.NET 内部对象和常用事件	311
12.5.1	什么是随机文件	265	习题 14		313
12.5.2	用户自定义类型	265	实验 14		313
12.5.3	打开和关闭随机文件	265	第 15 章	多媒体编程初步	314
12.5.4	读写随机文件	266	15.1	概述	314
12.6	使用文件系统对象	267	15.2	一个播放器的示例程序	314
12.6.1	My.Computer.FileSystem 对象属性	268	15.3	文字效果	315
12.6.2	My.Computer.FileSystem 对象方法	268	15.4	图像效果	317
12.7	程序举例	270	15.5	播放 WAV 声音文件	320
习题 12		273	15.6	AxWindowsMediaPlayer 多媒体 播放控件	322
实验 12		275	15.7	AxShockwaveFlash 控件	323
第 13 章	数据库的基本应用	276	习题 15		325
13.1	概述	276	实验 15		326
13.1.1	关系数据库	276	第 16 章	面向对象程序设计初步	327
13.1.2	创建数据库	277	16.1	概述	327
13.1.3	使用 SQL 语句	278	16.2	一个简单的面向对象程序设计例子	328
13.2	使用数据库方式验证用户名和密码 登录程序	279	16.3	类和对象	331
13.3	使用 ADO.NET 控件编写程序	280	16.3.1	类的创建	331
13.3.1	简单地用绑定控件显示数据	280	16.3.2	对象的创建和使用	333
13.3.2	使用 ADO.NET 表格数据控件	282	16.3.3	类的成员	333
13.3.3	导航栏与绑定控件	284	16.4	类的继承	340
13.4	使用 ADO.NET 对象访问数据库	285	16.4.1	继承的基本概念	340
13.4.1	ADO.NET 结构	285	16.4.2	继承的实现	340
13.4.2	Connection 对象	286	16.4.3	MyBase 和 MyClass	342
13.4.3	Command 对象	288	16.5	类的多态	343
13.4.4	DataReader 对象	289	16.6	异常处理方式	345
13.4.5	DataAdapter 对象	290	16.6.1	异常类	345
13.4.6	DataSet 对象	291	16.6.2	Try...Catch...Finally 语句	345
13.5	事务处理	293	16.6.3	自定义异常类及抛出异常	349
13.6	程序举例	294	16.7	LINQ 初步	350
习题 13		295	习题 16		352
实验 13		296	实验 16		353
第 14 章	编写简单的网络程序	297	第 17 章	综合程序设计举例	355
14.1	概述	297	17.1	个人通讯录	355
14.2	一个简单的网络浏览器	297	17.1.1	数据库设计	355
14.3	使用 WebBrowser 控件	298	17.1.2	系统实现	356
14.4	Socket 类的通信	301	17.2	我的记事本	360
14.4.1	Socket 类的通信协议	302	17.2.1	程序分析	360
14.4.2	创建 Socket 实例	302	17.2.2	程序实现	361
14.5	Visual Basic Script 简介	305	参考文献		367

第1章

概 述

本章首先回答为什么要学习程序设计,然后介绍有关程序设计语言的一般知识,并讨论有关编程的一些基本问题。本章通过对 Visual Basic.NET (简称 VB.NET) 历史和简单例子的介绍,使读者对 VB.NET 程序设计语言有个初步的认识。通过对结构化的编程方法、事件驱动编程和面向对象编程等技术的介绍,帮助读者理解通过计算机运行程序来解决应用问题的过程。

1.1 为何要学习程序设计

良好能力性的素质,往往表现在从问题的定义出发,明确问题的性质,发现问题的本质,找到解决问题的途径,并确定一种最好的处理方法,使得问题最终得以解决。无论是社会活动还是日常问题,无论是科学实验还是制定研究规则,这个过程总是类似的。

如果着眼于发现问题、解决问题的观点,学习计算机语言并尝试进行程序设计就是一种非常好的方法。这里首先需要理解“为什么要学习程序设计”,这非常重要:如果你不知道为什么要做某件事,很难想象你会把某件事做好。

1.1.1 什么是程序设计

大家知道,计算机依靠程序才能够工作。一般认为,程序(Program)是产生一系列指令(或者叫做命令)让计算机工作的过程,它们可以存储在计算机中,而这些指令的集合就是计算机程序设计语言。因此在这个意义上,程序设计有两个重要的思想:一是我们需要把复杂的设计过程翻译为机器能够理解的执行代码,二是程序被存储在计算机中可以反复地被执行。

现在有数以百计用于不同应用的计算机程序设计语言(Programming Language),这些语言大致可以分为三类:机器语言、汇编语言和高级语言。

机器语言(Machine Language)也叫做机器代码语言,使用二进制代码,计算机能够直接识别并执行这些代码,其中的代码又称为机器指令。汇编语言(Assemble Language)以英文助记符的形式表示,最后需要将其翻译为二进制代码才可运行。这两种语言都是面向机器的,也就是说,

不同 CPU 类型的计算机有不同的指令系统和汇编语言。使用机器语言、汇编语言的用户需要对所编程机器的指令系统有非常深入的掌握,所以一般被认为这是专业人员所从事的事。

随着计算机技术的发展,许多过去必须由计算机专业编程人员才能够从事的程序设计工作被更多的非专业人员所掌握。打破专业人员对计算机编程的“垄断”,应该归功于“高级语言”。

编程人员编写机器语言程序需要仔细地书写一个个 0 和 1 组成的代码,使用汇编语言则需要将每条指令用一个助记符代替,高级语言程序则可以简单地使用类似于数学表达或者英语单词组成其语句。高级语言可以让编程人员在较短的工作时间内完成更多的编程工作,大大提高了编程的效率。

1.1.2 有哪几种常用的高级语言

现在尽管有多种高级语言,但发展到今天能够称得上被广泛使用的高级语言也只有少数几种。

① C 语言:功能强大,一直有“专业语言”的美誉。C 语言由 Bell 实验室于 1972 年开发,兼有高级程序设计语言的优点和汇编语言的高效率,能够直接操纵计算机内存单元中的数据位,因此在设计系统软件(如操作系统)或者实时应用领域用途很广。使用 C 语言比使用其他高级语言写出的程序运行速度快得多。

② C++语言:C 语言的扩展版本。C++提供了若干特性,改善了 C 语言的功能,并增加了面向对象编程的功能。在今天的系统设计领域,C++已经成为主要设计语言之一。

③ FORTRAN 语言:在数学、科学计算等领域,FORTRAN (FORmula TRANslator,公式翻译)语言具有独特的优势。它也是最早的计算机高级编程语言,最初由 IBM 公司于 1957 年开发。

④ Java 语言:被称为网络编程语言,是面向对象的程序设计语言,是基于 C 和 C++的。Java 特别适合于 Internet 环境,现已成为 Internet 中最受欢迎、最有影响的编程语言之一。Java 的 Windows 版本叫做 Visual J++,很多人认为 Visual J++是 Visual Basic 的最有力竞争者。Java 是由美国 SUN 公司开发并于 1995 年发布的。

上面列举的几种,加上本书所介绍的 VB.NET,被认为是目前最广泛使用的几种编程语言。机器语言、汇编语言都是面向机器的,它们必须针对特定的机器进行程序设计,而高级语言则灵活得多,一般情况下都是与机器无关的。

1.1.3 翻译计算机语言程序

高级语言编写的程序(称为源程序)需要翻译为机器语言程序才能被计算机所执行。语言翻译程序也是一种软件,人们称之为“程序的程序”,根据其翻译的方式,可分为解释程序和编译程序。

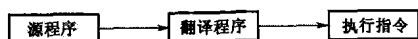


图 1.1 解释程序

解释程序对源程序中的代码翻译一句执行一句,如图 1.1 所示,这个过程有点像同声翻译。解释程序在执行

过程中不生成可执行的文件,因此程序运行速度较慢,而且源程序必须在解释程序环境下运行,如果源程序出错,它将立即停止执行。计算机使用解释程序可以在较小的内存中运行程序。早期的 BASIC 语言就是解释型语言,Visual Basic 保留了这个特点,同时使用编译方式。

编译程序的结构如图 1.2 所示,图中的虚线部分为语言处理程序。首先由编译器(Compiler)将源程序整体翻译为目标程序代码,然后由连接程序(Linker)将目标程序和系统资源(如函数库、系统过程等)连接在一起形成可执行程序文件,最终可以直接运行可执行程序文件,使得用户编写的程序脱离编译系统而独立运行。这个结果有点像外文文献的书面翻译。

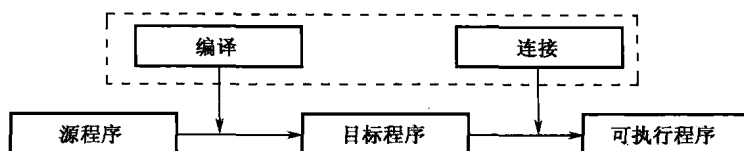


图 1.2 编译程序

在计算机文件中，可执行程序（Executable Program）文件大多以.exe 为后缀（扩展名），因此很好辨认。现在，许多编译系统已经直接将编译和连接过程组合在一起“构造”（Builder）可执行程序。Windows 系统将这类可执行程序文件统称为“应用程序文件”。

1.1.4 程序设计是一种方法学

一个很有意思的问题经常会被提及，那就是“为什么要学程序设计”。至少有一点是很明确的——几乎每个不同职业的人都要用到计算机软件：学生使用字处理软件撰写论文，会计师使用软件记账，摄影师使用图片编辑软件处理照片，电视台使用软件编辑节目，作家使用软件写作……事实上计算机是职业人员应该使用的工具，而理解计算机，学习编程是最好的途径。

学习程序设计和学习数学一样，能够培养一个人发现问题、处理问题的能力，而且这种训练更符合信息时代的发展要求。与传统课程不同的是，程序设计是一个主动的、积极的过程，因为在编程过程中计算机能够立即、直接地反馈信息，这需要编程者与之互动，达到探索、实验和评价的效果。若把编程与做数学习题的过程进行比较，就会发现程序设计更有趣、更生动：编程者与计算机直接交互，如计算机会告诉你某个语句出现了错误，你就必须检查并纠正这个错误，这是一个探索过程；同时，编程者有一种成就感，如果你的程序能被正确执行的话。

学过程序设计的人未必就一定要从事程序设计工作。事实上，目前编程的人越来越少，用程序的人越来越多，这就使得有人怀疑是否有学习程序设计的必要。其实回答这个疑问并不难，就像我们在大学里人人都学习微积分，但并没有几个人使用微积分。那么是不是不用的人就不需要学呢？如果你不以程序设计本身为目的进行学习，那么我们看看下面的事实。

在我们使用各种不同的软件帮助处理问题时，我们需要与计算机进行交互，如编辑文档、使用电子表格、制作演示文稿等。一个简单的例子是，如果你现在通过网络的搜索引擎去查找需要的资料，一个更符合检索要求的表达式是重要的，因此你理解计算机如何用程序来运行你的检索表达式，对你进行检索操作肯定会有帮助。如果你知道了计算机是如何用程序进行工作的，则你在电子表格中对符合条件的数据进行统计计数时，绝对不会一个个地把它数出来。今天我们使用的是这些软件，我们不能知道几年后将会有哪些软件，会帮我们解决哪些问题。因此，理解编程比直接理解使用软件有更长远的意义。

让计算机帮助人们做事，采用了人在与计算机交互过程中使用的特定科学符号，有些是自然语言表达，有些是具体的特定记号，这些都是某种形式的“编程”过程。即使使用某些功能很强大的软件包，如进行科学计算和工程计算的 MATLAB 之类的软件，仍然需要在某种程度上按照一定的表达形式进行“编程”，这些软件才能够帮助用户完成工作。

1.1.5 如何学习程序设计

一个程序员的工作与作家、设计师没有什么不同，都是从构思框架开始，然后进入细节，最终将其设计思想表现为特定的文字或者图纸。其中将根据故事的发展或设计要求的变化，反

复修改。

事实上,没有任何一个程序员设计程序能够一气呵成,都需要经过多次的反复修改,最后达到程序功能的要求。进一步,即使程序看上去正确,其中仍然会隐藏着未知的错误。因此,测试程序需要使用不同的输入,在尽可能多的环境下运行,以使得这些错误发生在程序正式投入运行之前,然后及时纠正。

几乎所有的程序设计都会涉及输入数据、处理数据,进一步地说,设计针对某个特定问题求解的程序,必须知道需要哪些数据、这些数据的性质(正式的说法是数据的属性),以及这些数据之间的相互关系。因此,要学习程序设计就必须学习表达这些数据以及相互关系的语言的描述,一般认为,需要:

- ① 分析问题,使用特定的方法进行描述,如使用流程图描述。
- ② 在表达抽象问题时,用明确的实例加以说明。
- ③ 使用编程语言精确描述所定义的语句和数据之间的关系。
- ④ 通过检查,包括机器编译过程的检查,测试上述设计。
- ⑤ 注重细节问题。

第一步的分析过程是至关重要的。如果说其后的步骤和计算机语言关系密切的话,那么分析问题则与“算法”相关。大多数处理过程还与数学表达有关,即选择一个合适的算法是程序设计的第一步。讨论“算法”不是本书的范围,但本书给出的例子都含有算法的影子,这需要读者从中领会。

以上这些表现在计算机程序设计中的行为,与任何一项有意义的事务(如一个商业活动)的过程都是类似的。程序设计过程与一般处理问题的过程有一个重要的差别,即衡量程序的设计结果不是靠设计者或使用者的评价,更重要的是机器的评价,即程序必须被正确地运行,至少在预见的范围内,程序的运行结果能够与预期结果相同。

因此,强调实践环节有时比设计过程的学习更要紧。传统的阅读理解是需要的,但有时把设计的代码输入到计算机,让计算机执行一次再来查看程序的执行结果,比仅靠阅读来理解程序更有效,也就是说,实验能够帮助我们进一步理解程序设计的过程。

1.2 什么是 Visual Basic.NET

Basic 正确的书写应该是 BASIC,因为它是 Beginners All-purpose Symbolic Instruction Code(初学者通用指令码)的缩写。实如其名,BASIC 的语法规则相对简单,易于理解和掌握,被许多非计算机专业编程人员和爱好者所喜欢,也被广泛使用。最初的 PC 系统内置的软件就是 BASIC,因此可以认为 BASIC 是与 PC 的发展同步的。

早先的微机系统是字符界面的,而现在广泛使用的主要是基于 GUI (Graphics User Interface) 的 Windows 系统。20 世纪 90 年代以来,Windows 应用软件就逐渐成为程序设计的主流。而开发 Windows 程序,大量的编程工作是建立交互界面,如窗口、菜单、对话框等,使得程序开发工作量大大增加。据统计,在一个 Windows 应用程序中大概有 80%的工作是设计这些交互界面。

计算机系统技术的这些发展使得基于 Windows 的 GUI 环境的程序设计语言随之发展,因此 BASIC 进化为 Visual Basic,它是 Microsoft 公司于 1991 年开发的。

Visual Basic 是随 Windows 发展而发展的,因此它被称为“Windows 程序设计语言”。从这个意思上看,学习 Visual Basic 的确是“容易”的。因为它把 Windows 的大多数资源,如窗口、菜

单、对话框、按钮等“对象”集成起来，纳入到 Visual Basic 中，程序员需要使用这些对象时，可以简单地“拿来”使用，而不必自己再花大量的时间编写这些对象的代码，可以集中精力编写程序对数据的处理过程。Visual Basic 给这些“对象”取了专用的名字：控件（Control）。

继 Visual Basic 之后，微软公司推出了全新的“.NET 框架”。.NET 是微软的新一代技术平台，为敏捷商务构建互连互通的应用系统，这些系统是基于标准的、连通的、适应变化的、稳定的和高性能的。从技术的角度来看，一个.NET 应用是一个运行于.NET Framework 之上的应用程序。.NET Framework（.NET 框架）由微软开发，是致力于敏捷软件开发、快速应用开发、平台无关性和网络透明化的一个软件开发平台。

.NET 框架采用系统虚拟机运行的编程平台，以通用语言运行库（Common Language Runtime, CLR）为基础，支持 C#、C++、Python 等多种语言的开发，其中包含了 Visual Basic.NET，它是这个系统中的重要部分之一。

Visual Basic.NET 的简称为 VB.NET。Visual Basic 只是半面向对象的编程环境，其面向对象的能力及执行效率往往不能满足一些大项目的需要，所以 Visual Basic 的最后一个版本是 Visual Basic 6.0。

在 Visual Basic 6.0 之后，微软第一个 Visual Studio .NET 7.0（Visual Studio 简称 VS）版本中集成了 Visual Basic 7.0、Visual C++ 7.0 等。Visual Basic 7.0（VB 7.0）是 VB.NET 的第一个版本，所以它是微软新平台的技术。VB.NET 编译以后生成的可执行文件被称为 Assembly，即程序集。本书使用的 VB.NET 的版本号是 Visual Basic 9.0，是 Visual Studio 2008 的组件之一。

设计程序有两个主要的部分：一是与用户交互部分的“界面设计”，二是对数据进行处理程序代码设计。界面呈现在计算机显示器屏幕上，VB.NET 把界面中的对象通过“控件”提供给编程人员，整个程序设计的界面就是“可见的”，这就是 Visual 的意思。

编写 Windows 环境下的应用程序，使用 VB.NET 是方便编程的选择之一。今天的 VB.NET 仍是许多专业人员选择的 Windows 应用系统开发工具之一。VB.NET 有强大的工具和丰富的控件，并提供了多媒体功能，适合于普通应用，再加上基于 Internet/Intranet 应用的编译代码，对软件业的发展和变革也起到了重要的作用。

1.3 VB.NET 平台：集成开发环境

本书介绍 VB.NET（也称为 Visual Basic 2008 或 VB 9.0），为方便起见，后面的章节中仍称其为 VB。编写一个应用程序的过程包括：创建程序、编辑代码、调试程序、运行程序等多个步骤。早期的编程是将这些步骤分别使用不同的系统（也是程序）来完成。

可以把平台视为一个舞台，不同的表演形式可以在相同的舞台上展示，那么 Windows 就是支持各种基于 Windows 操作系统的应用程序的运行舞台。VB.NET 也是在 Windows 环境下运行的。VB.NET 提供给编程者在同一个环境中建立或打开源程序、编辑代码、调试运行程序等，可完成所有的编程工作，这就是 IDE（Integrated Development Environment，集成开发环境）。因此，也可以说.NET 是一个包括多种语言且具有解释、编译 VB.NET 应用程序的综合型编程工具软件。

.NET 本身就是 Windows 的应用程序，它的许多资源如菜单、对话框、工具按钮也是被它自己所使用的，因此在.NET 环境下编写用户系统往往会有一种非常特别的感觉：似曾相识。

在计算机上安装 VS 2008 后，从桌面任务栏上选择“开始”→“所有程序”→“Microsoft Visual Studio 2008”→“Microsoft Visual Studio 2008”，系统将显示如图 1.3 所示的“选择默认环境设置”对话框，从中选择“Visual Basic 开发设置”，单击“启动 Visual Studio”按钮，这时 Microsoft Visual

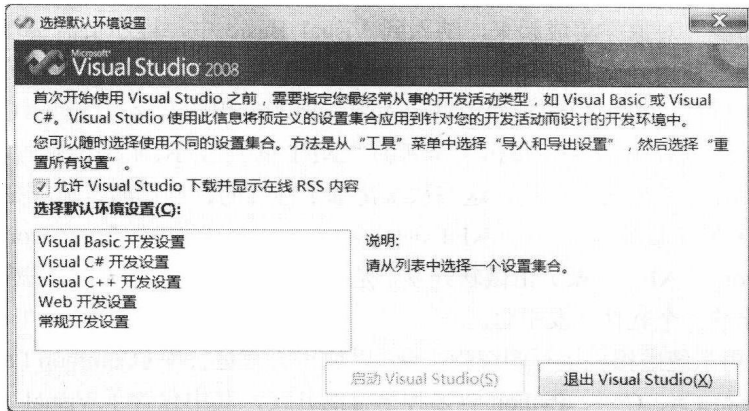


图 1.3 选择默认环境设置

Studio 为第一次使用配置环境。在等待几分钟后，进入如图 1.4 所示的运行界面，它是集成环境中默认的浏览器主页，也是选择最近使用的项目、读取产品新闻和了解 VB 新功能等操作的地方。

在图 1.4 所示的界面中，选择“文件”菜单的“新建项目”命令，出现如图 1.5 所示的“新建项目”对话框，编程者可选择使用哪一种编程语言来创建哪一种应用程序。

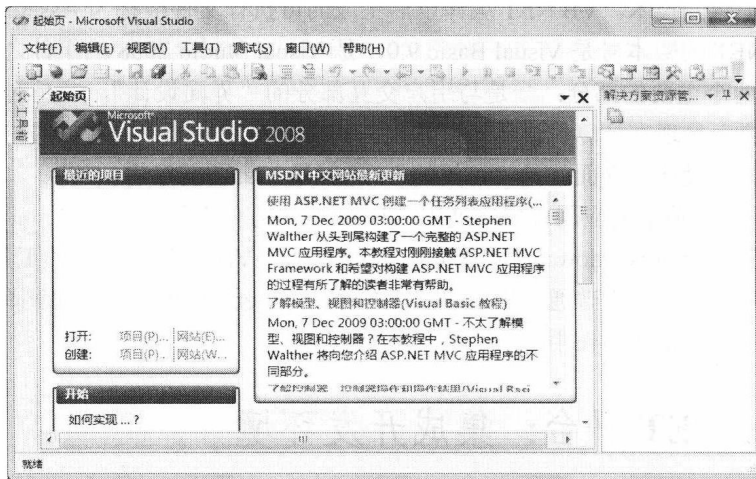


图 1.4 Microsoft Visual Studio 起始运行界面

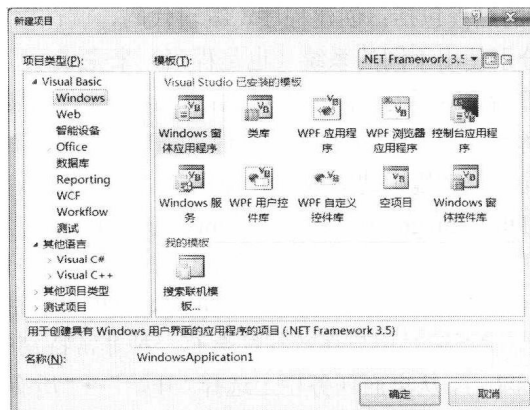


图 1.5 “新建项目”对话框

其中“Visual Basic”就有多种项目，每种项目又提供了多种应用程序模板，可见.NET 提供的编程功能非常强大。VB.NET 使用“项目”作为用户编程的代名词。利用图 1.5 所示对话框新建项目，VB.NET 就把用户选定的项目类型的相关特性装入，进入设计主窗口。

现在，选择项目类型为“Visual Basic”下的“Windows”，然后在“模板”中选择“Windows 窗体应用程序”，这是 Windows 类型下的默认模板，允许程序员使用 VB 的大多数特性，可以生成在 Windows 环境下直接执行的程序文件。本书的大多数示例程序和编程都使用这个模板。在该对话框的“名称”处输入项目名“Example”，单击“确定”按钮，这时出现如图 1.6 所示的 VB 窗体应用程序开发界面。

在图 1.6 所示窗口中，标题栏上已出现了目前项目名称“Example”，它是一个多窗口的操作环境，图中所示的是程序员常用的几个窗口，其中的工具箱如图 1.7 所示。这些窗口初次使用时自动打开，如果需要关闭或者重新打开，可在“视图”菜单中选择相应的选项。

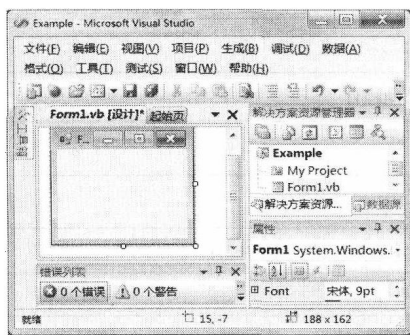


图 1.6 Visual Studio 窗口



图 1.7 工具箱

在图 1.6 所示窗口中，我们可以完成 VB.NET 的编程工作。这个窗口的基本操作（如菜单、工具按钮等）与 Windows 的窗口操作类似。有关它们的使用，我们将在后续章节中根据进度介绍，这里不再赘述。

其中的“帮助”菜单可以打开帮助窗口。“帮助”（快捷键 F1）提供 VB.NET 的功能、控件特性、函数使用等非常详细的介绍。无论用户如何熟悉 VB.NET，在线帮助仍然是需要的，因为用户不可能靠记忆掌握 VB.NET 各种元素的复杂特性和使用规则。

（1）窗体和窗体编辑器

在图 1.6 所示窗口中，“Form1.vb[设计]”选项卡中有标题为“Form1”的子窗口，它们分别是窗体编辑器和窗体。

“窗体”是 Form 的中文翻译，本意是外形或者形状。简单地说，在进行 VB.NET 编程时，窗体可以作为用户界面的载体（容器），存放程序中的按钮、菜单、各种标签、文本框等组件（控件）。

当所设计的程序被执行时，窗体就是显示在屏幕上的用户程序界面。与一般的 Windows 程序一样，窗体可以接受用户的数据输入，可以显示执行程序后的各种输出。

在进行界面设计时，窗体和窗体编辑器是组合在一起的。当界面设计完成需要进入代码编写时，双击窗体中的任何空白处就可以进入编码状态，这时出现标签名为“Form1.vb”的选项卡，其中就是程序代码编辑器。

（2）工具箱

工具箱（ToolBox）位于 IDE 左侧，当鼠标指针移到左侧文字“工具箱”上，就会出现如图 1.7 所示的工具箱窗口。工具箱中有多组工具，如“公共控件”、“容器”等。图 1.7 所示即为