

教育部推荐教材

C#大学基础教程

主 编 黄 敏 傅 瑜
副主编 周新丰



NLIC 2970676201



北京师范大学出版集团
BEIJING NORMAL UNIVERSITY PUBLISHING GROUP
北京师范大学出版社

教育部推荐教材

C#大学基础教程

主 编 黄 敏 傅 瑜

副主编 周新丰

参 编 康汉彬



NLIC 2970676201



北京师范大学出版集团
BEIJING NORMAL UNIVERSITY PUBLISHING GROUP
北京师范大学出版社

图书在版编目(CIP)数据

C#大学基础教程 / 黄敏, 傅瑜主编. —北京: 北京师范大学出版社, 2010.8

ISBN 978-7-303-11170-1

I. ① C… II. ①黄… ②傅… III. ① C语言—程序设计—高等学校: 技术学校—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字(2010)第 126641 号

出版发行: 北京师范大学出版社 www.bnup.com.cn

北京新街口外大街 19 号

邮政编码: 100875

印 刷: 北京嘉实印刷有限公司

经 销: 全国新华书店

开 本: 184 mm × 260 mm

印 张: 13.75

字 数: 269 千字

版 次: 2010 年 8 月第 1 版

印 次: 2010 年 8 月第 1 次印刷

定 价: 23.00 元

策划编辑: 周光明

责任编辑: 周光明

美术编辑: 高 霞

装帧设计: 华鲁印联

责任校对: 李 菡

责任印制: 李 啸

版权所有 侵权必究

反盗版、侵权举报电话: 010-58800697

北京读者服务部电话: 010-58808104

外埠邮购电话: 010-58808083

本书如有印装质量问题, 请与印制管理部联系调换。

印制管理部电话: 010-58800825

前 言

C#语言是 .Net Framework 中新一代的开发工具,已成为在 .NET 平台上进行开发的首选语言。Visual C# 2005 将 C++ 的灵活性和强大功能与 Visual Basic 的简单性融于一体,不但如此,C#还能为 C++ 程序员提供快捷的开发方式,又没有丢掉 C 和 C++ 的基本特征——强大的控制能力。C# 与 C 和 C++ 有着很大程度上的相似性,熟悉 C 和 C++ 的开发人员很快就能精通 C#。

本书采用案例教学的方法,内容力求简明、通俗,案例贴近现实生活。配套有全套 PPT 课件、上机指导书、习题解答、教学大纲及完整的程序源代码,方便教师教学、读者自学。特别适合作普通本科院校、独立学院、高职高专的程序设计课程教材。

书中总共讲述了 12 章的内容。前 7 章为 C# 基础部分,简单讲述了通过 C# 编写 Windows 控制台程序的一些基本知识,并着重突出 C# 区别于 C/C++ 的语言特色。第 8~11 章为 Windows 窗体部分,着重简述了常用 Windows 控件的使用方法、菜单设计、GDI+ 编程、数据库与 ADO.NET 这几方面的内容。第 12 章为 ASP.NET 简介部分。书中知识点层次分明,便于读者学习与查阅。

本书由电子科技大学中山学院黄敏副教授、傅瑜教授统编和定稿,并担任主编;湖南潇湘职业技术学院周新丰老师担任副主编。参与本书编写工作的还有康汉彬老师。

在本书编写过程中,作者得到了很多同事的帮助,.NET 教学团队、李赞老师等人为本书提出了很多宝贵的意见和建议,陈烁同学做了大量的修订和程序验证工作,在此表示诚挚谢意。

本书的操作环境为 Visual Studio 2005,书中所有的实例都在此编译环境中运行通过。本书配有 PPT 教案、上机实验指导、部分习题解答、程序源代码,其中 PPT 教案、上机实验指导以及源代码可在北京师范大学出版社职教网站(www.zjfs.bnup.com.cn)下载,也可与出版社联系,电话 010-5880275。

由于时间仓促,加之编者水平有限,书中的缺点和不足之处在所难免,敬请读者批评指正。

作者的电子邮箱地址:hm0607@sina.com.cn 或 8055509@qq.com

编 者
2010 年年初

《面向对象程序设计(C#)》课程实验教学大纲

课程编号： 适用专业：计算机科学与技术、软件工程、网络工程等
 授课学时：32 学时 学分数：3
 先修课程：《计算机基础》、《C++面向对象程序设计》、《数据库基础》
 考核方式：系考

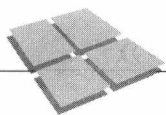
一、实验教学目的和任务

《面向对象程序设计(C#)》是计算机专业的一门必修课程。面向对象软件开发方法是吸收了软件工程领域有益的概念和方法而发展起来的一种软件开发方法。它集封装性、抽象性、继承性与多态性于一体，可以帮助人们开发出模块化、数据抽象程度高的，体现信息隐蔽、可复用性、易修改、易扩充等特性的程序。本课程主要介绍面向对象程序设计方法与思想。以C#语言面向对象机制为主，使学生的学习过程中通过大量的程序实例和相关练习，逐步掌握C#的面向对象的程序设计方法与思想，从而掌握面向对象程序设计的基本知识和基本技能，使学生掌握面向对象程序设计的方法与原理，会编写和使用类，掌握常用控件的使用，熟悉使用ADO.NET进行数据库编程，能够熟练地使用Visual Studio.NET集成开发工具编写Windows应用程序和数据库应用程序。

本实验大纲力求结合该课程教学计划大纲的相应内容，由浅入深的指导学生了解和掌握如何利用C#语言进行程序设计，提高学生的动手能力，做到理论和实践相结合，培养学生理解、分析程序，编写、调试程序的能力，使之能把程序设计应用到今后的专业学习中。通过实验加深对基本概念和基本C#语法点的掌握深入掌握Windows应用程序开发、GDI+编程、数据库编程、ASP.NET网站开发等知识。

二、实验项目名称及学时分配

实验项目和名称	实训教学课时数	实验类型
.NET与C#概述	2	验证
C#数据类型与流程控制	4	验证
面向对象程序设计	6	综合
常用Windows控件	6	验证
菜单设计	2	设计
GDI+编程	4	设计
数据库与ADO.NET	6	设计
ASP.NET简介及综合测评	2	综合
合计	32	32



三、实验教学内容和要求

实验内容：见上机指导书。

实验要求：

实验一 .NET 与 C# 概述(2 学时)

(1) 了解和使用 VS2005 集成开发环境。

(2) 理解简单的 C# 程序结构。

实验二 C# 数据类型与流程控制(4 学时)

(1) 理解各种数据类型的概念、掌握变量的定义。

(2) 理解选择结构语句。

(3) 熟练掌握 while 语句, do... while 语句和 for 语句实现循环的方法。

(4) 能够灵活运用各种循环控制进行综合程序设计。

(5) 掌握在程序设计中用循环的方法实现各种算法(如穷举等)。

实验三 面向对象程序设计(6 学时)

(1) 掌握类的概念、类的定义格式、类与结构的关系、类的成员属性和类的封装性。

(2) 掌握类对象的定义。

(3) 理解类的成员的访问控制的含义, 公有、私有和保护成员的区别。

(4) 掌握构造函数和析构函数的含义与作用、定义方式和实现, 能够根据要求正确定义和重载构造函数。能够根据给定的要求定义类并实现类的成员函数。

(5) 掌握继承与派生、类的多态性的含义, 熟练应用其进行编程。

(6) 了解文件操作的基本概念。

(7) System. IO 命名空间中最重要的一些类, 通过运用这些类我们就能完成基本的文件操作。

实验四 常用 Windows 控件(6 学时)

(1) 掌握文本类控件和动作类控件的运用。

(2) 掌握列表类控件进行 windows forms 应用程序的开发。

(3) 掌握值类控件和图像类控件进行 windows forms 应用程序的开发。

(4) 掌握日期类控件和其他类型控件进行 windows forms 应用程序的开发。

(5) 能熟练运用 Windows 控件进行一些综合实例的编程。

实验五 菜单设计(2 学时)

(1) 掌握文本编辑器的基本构架。

(2) 了解菜单、工具栏、状态栏设计的基本方法。

(3) 实现文本编辑器中菜单栏和工具栏的基本功能。

实验六 GDI+ 编程(4 学时)

(1) 掌握图形设计的基本工具的使用方法。

(2) 能用 graphics 对象进行简单的一般性的图形设计与文本显示。

实验七 数据库与 ADO.NET(6 学时)

- (1)了解如何创建和删除数据库。
- (2)掌握连接数据库对象(Connection)。
- (3)掌握执行 sql 语句对象(Command)。
- (4)掌握读取数据对象(DataReader)。
- (5)掌握数据适配器对象(DataAdapter)。
- (6)掌握数据集对象(DataSet)。
- (7)了解 BindingSource 组件和 DataGridView 数据控件。
- (8)通过 BindingSource 组件和 DataGridView 数据控件可以直接对数据进行操作，增强操作数据库中数据的效率。

实验八 ASP.NET 简介及综合测评(2 学时)

- (1)通过 .NET 编程语言 C# 和 .NET 调试功能创建 ASP.NET Web 应用程序。
- (2)综合测评。

四、教材和参考资料

1. 建议教材

郑阿奇，梁敬东著. C# 语言程序设计教程. 北京：机械工业出版社，2008

2. 参考书目

①李兰友，杨晓光. Visual C#.NET 程序设计. 北京：清华大学出版社，2004

②索利斯 (Daniel Solis)，苏林，朱晔. C# 图解教程(图灵程序设计丛书，微软技术系列). 北京：人民邮电出版社，2009

编著

2010 年 3 月

《面向对象程序设计(C#)》课程理论教学大纲

课程编号： 适用专业：计算机科学与技术、软件工程、网络工程等
授课学时：32 学时 学分数：3
先修课程：《计算机基础》、《C++面向对象程序设计》、《数据库基础》
考核类型：系考

一、课程性质和目的

C# 语言是 .NET Framework 中新一代的开发工具，是计算机类各专业的一门重要的专业基础课。本课程将全面、系统地介绍 C# 的基本知识和程序设计的基本方法。

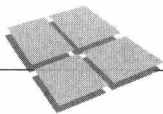
《面向对象程序设计(C#)》是计算机专业的一门必修课程。面向对象软件开发方法是吸收了软件工程领域有益的概念和方法而发展起来的一种软件开发方法。它集封装性、抽象性、继承性与多态性于一体，可以帮助人们开发出模块化、数据抽象程度高的，体现信息隐蔽、可复用性、易修改、易扩充等特性的程序。本课程主要介绍面向对象程序设计方法与思想。以 C# 语言面向对象机制为主，使学生在在学习过程中通过大量的程序实例和相关练习，逐步掌握 C# 的面向对象的程序设计方法与思想，从而掌握面向对象程序设计的基本知识和基本技能，使学生掌握面向对象程序设计的方法与原理，会编写和使用类，掌握常用控件的使用，熟悉使用 ADO.NET 进行数据库编程，能够熟练地使用 Visual Studio .NET 集成开发工具编写 Windows 应用程序和数据库应用程序。

二、教学内容和要求

课程的主要内容：

学时数和学时分配：

知识模块	理论教学课时数	实训教学课时数
.NET 与 C# 概述	2	2
C# 数据类型与流程控制	4	4
面向对象程序设计	6	6
常用 Windows 控件	6	6
菜单设计	2	2
GDI+ 编程	4	4
数据库与 ADO.NET	6	6
ASP.NET 简介	2	2
合计	32	32



课程内容以介绍面向对象程序设计的思想和方法为主,重点介绍面向对象程序设计的基本知识和基本技能,使学生能够熟练地使用 Visual Studio .NET 集成开发工具编写 Windows 应用程序和数据库应用程序。其先修课程为《计算机基础》、《C++ 面向对象程序设计》、《数据库基础》,在数据结构(C# 描述)、ADO.NET、ASP.NET 网络编程、软件工程等课程上要用到本课程相关知识。

本书重点放在面向对象程序设计方法、菜单设计、图形图像处理及 ADO.NET 数据库编程上,难点主要集中于 ADO.NET 综合编程。处理的方法主要采用流程图的方式讲清重点,并借助于生动的案例教学和上机实验攻克难点。

编写采用任务驱动——案例教学方法,内容由浅入深,易懂易学,案例精典,注重实战。

本课程内容会出现在数据结构(C# 描述)、ADO.NET、ASP.NET 网络编程、软件工程等课程中,处理的方法本着够用、实用、重应用为主。

课程的基本要求:

本课程主要介绍 C# 语言的基本概念和程序设计的方法。要求学习者在学习过程中通过大量的程序实例和相关练习,逐步掌握 C# 语言的面向过程的思想和方法,从而掌握使用 C# 语言进行程序设计的基本知识和基本技能。

实验和实践环节的安排:

实验安排:在专用机房安装 VS2005、SQL2005 软件,每周上机一次,有专门的实验老师指导。

实践环节安排:课程学完之后,安排两周左右时间进行课程设计。在老师指导下,学生以团队形式完成小型项目。

三、教材和参考资料

1. 建议教材

郑阿奇,梁敬东著.C# 语言程序设计教程.北京:机械工业出版社,2008

2. 参考书目

①李兰友,杨晓光.Visual C#.NET 程序设计.北京:清华大学出版社,2004

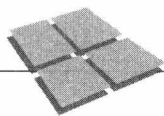
②索利斯(Daniel Solis),苏林,朱晔.C# 图解教程(图灵程序设计丛书,微软技术系列).北京:人民邮电出版社,2009

编著

2010 年 3 月

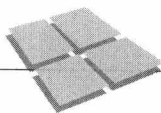
目 录

第 1 章 .NET 与 C# 简介	(1)
1.1 .NET Framework 介绍	(1)
1.2 C# 语言介绍	(1)
1.3 C# 语言特点	(2)
1.4 习题	(3)
第 2 章 C# 编程概述	(4)
2.1 一个简单的 C# 程序	(4)
2.2 如何建立控制台应用程序	(4)
2.3 程序的输入/输出	(5)
2.4 编程的规范	(6)
2.4.1 命名规范	(6)
2.4.2 编码规范	(7)
2.4.3 编码风格规范	(7)
2.4.4 注释规范	(7)
2.5 习题	(9)
第 3 章 变量、常量与运算符	(10)
3.1 变量和常量	(10)
3.1.1 常量	(10)
3.1.2 变量	(12)
3.2 运算符	(12)
3.2.1 算术运算符	(13)
3.2.2 关系运算符	(13)
3.2.3 逻辑运算符	(14)
3.2.4 位运算符	(14)
3.2.5 赋值运算符	(15)
3.2.6 条件运算符	(16)
3.2.7 运算符的优先级与结合性	(16)
3.3 习题	(16)
第 4 章 数据类型	(18)
4.1 值类型	(18)
4.1.1 简单类型	(18)



4.1.2	枚举类型	(19)
4.1.3	结构类型	(20)
4.2	引用类型	(21)
4.2.1	object 类	(21)
4.2.2	数组类	(21)
4.2.3	字符串类	(22)
4.3	值类型与引用类型关系	(23)
4.4	习题	(24)
第 5 章	控制语句	(26)
5.1	算法概述	(26)
5.2	选择语句	(26)
5.2.1	if 语句	(27)
5.2.2	switch 语句	(31)
5.3	循环语句	(33)
5.3.1	while 语句	(33)
5.3.2	do_while 语句	(34)
5.3.3	for 语句	(35)
5.3.4	三种循环的比较	(37)
5.4	跳转语句	(37)
5.4.1	break 语句	(37)
5.4.2	continue 语句	(37)
5.4.3	return 语句	(38)
5.4.4	goto 语句	(39)
5.5	foreach 语句	(40)
5.6	异常语句	(41)
5.7	习题	(43)
第 6 章	面向对象程序设计	(44)
6.1	面向对象程序设计概念	(44)
6.2	类与对象	(44)
6.2.1	类与对象概述	(44)
6.2.2	面向对象程序设计相关概念	(44)
6.2.3	类的声明	(45)
6.2.4	类的声明与对象的实例化	(46)
6.2.5	类的成员	(46)
6.3	继承与派生	(48)
6.3.1	基类和派生类	(48)
6.3.2	base 关键字	(50)
6.3.3	C# 语言类继承特点	(51)
6.4	类的多态性	(51)

6.5	委托与事件	(53)
6.5.1	委托	(53)
6.5.2	事件	(55)
6.6	接口	(57)
6.7	属性	(58)
6.8	习题	(60)
第7章	文件操作	(62)
7.1	概述	(62)
7.2	文件读写	(62)
7.3	习题	(64)
第8章	Windows 控件	(65)
8.1	控件概述	(65)
8.2	文本类控件	(65)
8.2.1	Label 控件	(65)
8.2.2	LinkLabel 控件	(67)
8.2.3	TextBox 控件	(69)
8.2.4	RichTextBox 控件	(72)
8.3	动作类型控件	(73)
8.4	列表类型控件	(76)
8.4.1	ListBox 控件	(76)
8.4.2	ComboBox 控件	(79)
8.4.3	ListView 控件	(81)
8.4.4	TreeView 控件	(85)
8.5	值类型控件	(87)
8.5.1	CheckBox 控件	(87)
8.5.2	RadioButton 控件	(90)
8.5.3	CheckedListBox 控件	(93)
8.6	图形类控件	(96)
8.7	日期类控件	(98)
8.7.1	DateTimePicker 控件	(98)
8.7.2	MonthCalendar 控件	(100)
8.8	其他类型控件	(103)
8.8.1	ToopTip 控件	(103)
8.8.2	ProgressBar 控件	(104)
8.8.3	WebBrowser 控件	(105)
8.8.4	MenuStrip 控件	(107)
8.8.5	GroupBox 控件	(109)
8.9	综合实例	(111)
8.9.1	高级计算器	(111)



8.9.2 浏览器	(112)
8.10 习题	(113)
第 9 章 菜单设计	(115)
9.1 菜单	(115)
9.1.1 菜单的概念和用途	(115)
9.1.2 菜单的设计	(115)
9.2 工具栏	(117)
9.2.1 工具栏用途	(117)
9.2.2 工具栏的设计	(117)
9.3 状态栏	(119)
9.3.1 状态栏介绍	(119)
9.3.2 状态栏设计	(119)
9.4 剪切板功能	(121)
9.5 存取文件	(122)
9.6 修改字体	(125)
9.7 综合实例	(127)
9.8 习题	(128)
第 10 章 GDI+ 编程	(129)
10.1 GDI+ 概述	(129)
10.2 创建 Graphics 对象	(130)
10.3 颜色和画笔	(131)
10.4 画刷	(133)
10.5 基本图形的绘制和填充	(139)
10.5.1 绘制矩形	(139)
10.5.2 绘制椭圆	(140)
10.5.3 绘制圆弧	(144)
10.5.4 绘制多边形	(145)
10.5.5 绘制基数曲线	(149)
10.6 综合实例	(150)
10.7 习题	(151)
第 11 章 数据库与 ADO.NET	(152)
11.1 概述	(152)
11.2 Connection 对象	(154)
11.3 Command 对象	(159)
11.4 DataReader 对象	(162)
11.5 DataSet 对象与 DataAdapter 对象	(164)
11.6 数据绑定	(166)
11.6.1 BindingSource 组件	(166)
11.6.2 DataGridView 控件	(168)

11.7	综合实例	(168)
11.8	习题	(170)
第 12 章	ASP.NET 简介	(172)
12.1	ASP.NET 概述	(172)
12.2	HTML 与 XHTML	(173)
12.2.1	HTML 和 XHTML 概述	(173)
12.2.2	HTML 和 XHTML 的格式与标记	(174)
12.3	ASP.NET WEB 窗体	(178)
12.4	ASP.NET 内建对象	(180)
12.4.1	Response 和 Request 对象	(180)
12.4.2	Cookie 对象	(183)
12.4.3	Application 对象	(184)
12.4.4	Session 对象	(184)
12.5	综合实例	(186)
12.6	习题	(186)
附 录		(188)
附录一	SQL 速成	(188)
附录二	HTML 标签详解	(191)
习题解答		(197)
参考文献		(202)

第 1 章 .NET 与 C# 简介

1.1 .NET Framework 介绍

Microsoft .NET(简称 .NET)是微软于 2000 年 6 月 22 日推出的战略核心产品,到了 2002 年, .NET 由计划成为现实,为广大的软件开发人员提供了未来计算的高效 Web Service 开发工具。 .NET 运行平台称为 .NET 框架(.NET Framework),是 .NET 平台的基础架构,它创造了一个完全可操控的安全的和特性丰富的应用执行环境,这不但使得应用程序的开发与发布更加简单,并且实现了众多语言间的无缝集成。

.NET Framework 是支持生成和运行下一代应用程序和 XML Web Services 的内部 Windows 组件,是 .NET 战略的核心。 .NET Framework 的目标是:

(1)提供一个一致的面向对象的编程环境,而无论对象代码是在本地存储和执行,还是在本地执行但在 Internet 上发布,或者是在远程执行。

(2)提供一个将软件部署和版本控制冲突最小化的代码执行环境。

(3)提供一个可提高代码(包括由未知的或不完全受信任的第三方创建的代码)执行安全性的代码执行环境。

(4)提供一个可消除脚本环境或解释环境的性能问题的代码执行环境。

(5)使开发人员的经验在面对类型大不相同的应用程序(如基于 Windows 的应用程序和基于 Web 的应用程序)时保持一致。

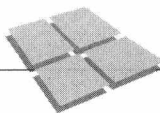
(6)按照工业标准生成所有通信,以确保基于 .NET Framework 的代码可与任何其他代码集成。

C# 语言是 .NET Framework 中新一代的开发工具,那么 C# 语言有什么优点,它为什么会成为现代的一种潮流,在下一节中我们会详细介绍。

1.2 C# 语言介绍

在最近的一段时间里,C 和 C++一直是最有生命力的程序设计语言。这两种语言为程序员提供了丰富的功能、高度的灵活性和强大的底层控制能力。而这一切都不在效率上作出不同程度的牺牲。如果你使用过包括 C 和 C++在内的多种程序设计语言,相信你会深刻体会到它们之间的区别。比如与 Visual Basic 相比,Visual C++程序员为实现同样的功能就要花费更长的开发周期。由于 C 和 C++既为我们带来了高度的灵活性,又使我们必须要忍受学习的艰苦和开发的长期性,许多 C 和 C++程序员一直在寻求一种新的语言,以图在开发能力和效率之间取得更好的平衡。

今天,人们改进、开发出了许多语言以提高软件生产率,但这些或多或少都以牺牲 C 和 C++程序员所需要的灵活性为代价。这样的解决方案在程序员身上套上了太多的枷锁,限制了它们能力的发挥。它们不能很好地与原有的系统兼容,更为令人头



痛的是它们并不总是与当前的 Web 应用结合得很好。

理想的解决方案，是将快速的应用开发与对底层平台所有功能的访问紧密结合在一起。程序员们需要一种环境，它与 Web 标准完全同步，并且具备与现存应用间方便地进行集成的能力。除此之外，程序员们喜欢它允许自己在需要时使用底层代码。

针对该问题，微软的解决方案是一种称之为 C# 的程序语言。C# 是一种现代的面向对象的程序开发语言，它使得程序员能够新的微软 .NET 平台上快速开发种类丰富的应用程序。.NET 平台提供了大量的工具和服务，能够最大限度地发掘和使用计算及通信能力。

由于其一流的面向对象的设计，从构建组件形式的高层商业对象到构造系统级应用程序，你都会发现，C# 将是最合适的选择。使用 C# 语言设计的组件能够用于 Web 服务，这样通过 Internet，可以被运行于任何操作系统上的任何编程语言所调用。

不但如此，C# 还能为 C++ 程序员提供快捷的开发方式，又没有丢掉 C 和 C++ 的基本特征——强大的控制能力。C# 与 C 和 C++ 有着很大程度上的相似性，熟悉 C 和 C++ 的开发人员很快就能精通 C#。

1.3 C# 语言特点

用 C# 语言编写的源程序，被 C# 语言编译器译为中间语言代码，形成扩展名为 .exe 或 .dll 文件。中间语言代码不是 CPU 可执行的机器码，在程序运行时，必须通过通用语言运行环境(Common Language Runtime, CLR)中的即时编译器(Just In Time, JIT)将中间语言代码翻译为 CPU 可执行的机器码，由 CPU 执行。CLR 为 C# 语言中间语言代码运行提供了一种运行环境，C# 语言的 CLR 和 Java 语言的虚拟机类似。这种执行方法使运行速度变慢，但带来其他一些好处，主要有以下几方面：

(1)通用语言规范(Common Language Specification, CLS): .NET 系统包括 C#、C++、VB 和 J# 语言，它们都遵守通用语言规范。任何程序设计语言只要遵守通用语言规范，其源程序都可编译为相同的中间语言代码，由 CLR 负责执行，这样的代码叫托管代码。只要为其他操作系统编制相应的 CLR，中间语言代码也可在其他系统中运行。

(2)自动内存管理：CLR 内建垃圾收集器，当变量实例的生命周期结束时，垃圾收集器负责回收不被使用的实例，防止其占用内存空间。不必像 C 和 C++ 语言，用语句在堆中建立的实例，必须用语句释放实例占用的内存空间。CLR 具有内存管理功能。

(3)交叉语言处理：由于任何遵守通用语言规范的程序设计语言源程序，都可编译成相同的中间语言代码，不同语言设计的组件，可以互相通用，可以从其他语言定义类派生出本语言的新类。由于中间语言代码由 CLR 负责执行，因此异常处理方法是一致的，这在调试一种语言调用另一种语言的子程序时，显得特别方便。

(4)更加安全：C# 语言不支持指针，一切内存的范围都必须通过对象的引用变量来实现，只允许访问内存中允许访问的部分，这就防止病毒程序使用非法指针访问私有成员，也避免指针的误操作产生的错误。

(5)完全面向对象：不像 C++ 语言既支持面向过程设计，又支持面向对象设计，

C# 语言是完全面向对象的，在 C# 语言中不再存在全局函数、全局变量，所有的函数、变量和常量都必须定义在类中，避免了命名冲突。C# 语言不支持多继承。

1.4 习题

1. 下列哪些语言可以用来开发 XML Web Service? () (多选)
 - A. C
 - B. C++
 - C. Java
 - D. C#
2. 简述 .NET Framework 基本构架及 .NET Framework 与 C# 语言的关系。
3. 简述 C# 语言的特点是什么? C# 语言是如何管理内存的?