

北京工业大学

本科课程教学大纲
Undergraduate Course Syllabi

信息科学技术学院

2020 版

目 录

| | |
|------------------------|---|
| “C#程序设计基础”课程教学大纲 | 1 |
|------------------------|---|

“C#程序设计基础”课程教学大纲

英文名称: Fundamentals of Programming Principle and Practice in C#

课程编号: 0004188

课程性质: 通识教育选修课

学分: 3.0

学时: 48

课程类别: 工程经济与项目管理 文化自信与艺术鉴赏 科学探索与创新发展
道德修养与身心健康 沟通表达与全球视野

面向对象: 理工科非计算机类专业第二、第三及第四年级学生

先修课程: 工程数学、大学英语

教材:

[1]罗福强 熊永福 杨剑 Visual C#.NET 程序设计教程 (第3版) 人民邮电出版社 2020.3

参考书:

[1] 本杰明•帕金斯 C#入门经典 (第8版) [M],北京: 清华大学出版社, 2019.01

[2] [美]哈里森·费隆(Harrison Ferrone) C#实践入门: 快捷学习 C#编程和 Unity 游戏开发, 清华大学出版社, 2020.5

[3] Mark J. Price C# 8.0 和 .NET Core 3.0 高级编程,清华大学出版社, 2020.03

[4] 唐大仕 C#程序设计教程 (第2版) 清华大学出版社 2018.01

[5] 李春葆、曾平、喻丹丹 C#程序设计教程 (第3版) 清华大学出版社,2015.10

一、课程简介

本课面向没有编程基础的学生,选择 Windows 操作系统平台上基于 .NET Framework 的面向网络开发的纯面向对象程序设计语言 C#作为教学语言,采用用案例驱动与知识驱动相结合的方式,讲授 C#语言的基础知识、面向对象编程基础、Windows 窗体应用程序和 Web 应用程序的开发方法,培养学生的应用开发平台结合工程设计问题设计出解决方案的实际开发能力,为日后产工作中从事科学探索与创新发展打下知识和技能基础。

二、课程地位与教学目标

课程地位: 本课程属于通识教育选修课,具有一定的专业基础和专业应用性质,它培养学生利用当代实际应用开发平台 Visual Studio 和 C#面向对象编程语言的关键性知识点,结合具体案例的开发过程体会程序设计的方法和开发过程,给学生提供参与设计实现具有适度规模的应用程序系统的机会,培养其工程意识和利用知识解决复杂工程问题的能力。

本课程对工程教育认证标准毕业要求中的下列要求具有相关性:

[1] 工程知识: 能够将数学、自然科学、工程基础和专业用于解决复杂工程问题。

[2] 问题分析: 能够应用数学、自然科学和工程科学的基本原理,识别、表达、并通过文献研究分析复杂工程问题,以获得有效结论。

[3] 设计/开发解决方案: 能够设计针对复杂工程问题的解决方案,设计满足特定需求的系统、单元(部件)或工艺流程,并能够在设计环节中体现创新意识,考虑社会、健康、

安全、法律、文化以及环境等因素。

[4] 研究：能够基于科学原理并采用科学方法对复杂工程问题进行研究，包括设计实验、分析与解释数据、并通过信息综合得到合理有效的结论。

[5] 使用现代工具：能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性。

[9] 个人和团队：能够在多学科背景下的团队中承担个体、团队成员以及负责人的角色。

[10] 沟通：能够就复杂工程问题与业界同行及社会公众进行有效沟通和交流，包括撰写报告和设计文稿、陈述发言、清晰表达或回应指令。并具备一定的国际视野，能够在跨文化背景下进行沟通和交流。

[12] 终身学习：具有自主学习和终身学习的意识，有不断学习和适应发展的能力。

教学目标：

Microsoft .NET 是微软公司为互联网环境下 PC 电脑和各类嵌入式智能设备的嵌入式系统的操作系统软件的解决方案，它建立在 IT 行业标准和 Internet 标准之上，提供开发（工具）、管理（服务器）、使用（构造服务和智能客户端）以及 XML Web 服务体验（丰富的用户体验），已经成为今天正在使用的 Microsoft Windows 应用程序、工具和服务器的基础支撑部分，同时，新产品不断扩展 XML Web 的服务能力以满足云计算等所有业务需求，C#语言是在这一架构上设计开发的最关键的程序设计语言，掌握.NET 技术及 C#编程知识是现代软件开发的一项基本技能，是理工科专业学生利用现代开发工具解决科学计算及构建分布式应用系统必备的基本知识之一。

通过对 C#编程语言的学习和项目实践训练，为理工类各专业学生奠定必要的基础，以便将来在工作中具备能力开发出应用程序来解决专业领域内生产及科研问题中的数据获取与信息加工呈现的设计需求，培养学生的实际动手能力和开发能力，满足大数据时代对高级专门人才的需求。

本课程的教学目标：

1、掌握 C#语言面向对象编程基础、掌握 Windows 窗体程序设计基础以及基本的 web 编程基础。

2、能够根据工程基础和专业设计 .Net 软件开发领域的复杂工程问题的解决方案；

3、能够根据用户需求确定计算机 .Net 软件系统或相关产品的设计目标。

4、能根据目标选取恰当的 C#、.Net 开发技术与设计流程，并确定系统、模块或流程的研发方案，体现创新意识。

5、能针对 .Net 软件开发领域的复杂工程问题，开发、选择与使用恰当的 C#、.Net 技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟。

6、具备从事基于 C#和 .Net 的软件开发所需的职业素养以及符合社会主义核心价值观的个人素质。

支撑的毕业能力项[1]、[3]、[5]，具体对应关系说明如下：

[1] 工程知识：能够将数学、自然科学、工程基础和专业设计用于解决复杂工程问题。

本课程目标 1：掌握 C#语言面向对象编程基础、掌握 Windows 窗体程序设计基础以

及基本的 web 编程基础。

本课程目标 2：能够根据工程基础和专业知识设计 .Net 软件开发领域的复杂工程问题的解决方案

[3] 设计/开发解决方案：能够设计针对复杂工程问题的解决方案，设计满足特定需求的系统、单元（部件）或工艺流程，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。

课程目标 3：能够根据用户需求确定计算机 .Net 软件系统或相关产品的设计目标。

课程目标 4：能根据目标选取恰当的 C#、.Net 开发技术与设计流程，并确定系统、模块或流程的研发方案，体现创新意识。

课程目标 6：具备从事基于 C#和.Net 的软件开发所需的职业素养以及符合社会主义核心价值观的个人素质。

[5] 使用现代工具：能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性。

课程目标 5：能针对 .Net 软件开发领域的复杂工程问题，开发、选择与使用恰当的 C#、.Net 技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟。

课程目标 6：具备从事基于 C#和.Net 的软件开发所需的职业素养以及符合社会主义核心价值观的个人素质。

本课程对毕业要求指标点达成的支撑情况，详见表 1。

表 1 课程目标与毕业要求指标点的对应关系

| 序号 | 课程目标 | 毕业要求指标点 | | | |
|----|--|---------|---|---|---|
| | | 1 | 3 | 5 | 9 |
| 1 | 掌握 C#语言面向对象编程基础、掌握 Windows 窗体程序设计基础以及基本的 web 编程基础。 | ● | ● | ⊙ | ⊙ |
| 2 | 能够根据工程基础和专业知识设计 .Net 软件开发领域的复杂工程问题的解决方案（了解面向对象分析与设计方法及 UML 常用符号的用途） | ⊙ | ● | ● | ⊙ |
| 3 | 能够根据用户需求确定计算机 .Net 软件系统或相关产品的设计目标（掌握 Windows 应用程序的结构及工作原理） | ⊙ | ● | ● | ◎ |
| 4 | 能根据目标选取恰当的 C#、.Net 开发技术与设计流程，并确定系统、模块或流程的研发方案，体现创新意识。（掌握 .NET FCL 的组成及单窗体和多窗体程序的组成及设计方法、Web 应用程序的初步设计） | ● | ● | ● | ⊙ |
| 5 | 能针对 .Net 软件开发领域的复杂工程问题，开发、选择与使用恰当的 C#、.Net 技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟。 | ⊙ | ● | ● | ⊙ |
| 6 | 具备从事基于 C#和.Net 的软件开发所需的职业素养以及符合社会主义核心价值观的个人素质。 | ● | ⊙ | ⊙ | ⊙ |

注：●：表示有强相关关系，◎：表示有一般相关关系，⊙：表示有弱相关关系

三、课程教学内容及要求

1、课程内容及要求

分章节列出课程教学内容及对应课程目标的支撑，详见表 2。

表 2 教学内容与课程目标的对应关系

| 章节名称 | 教学内容及重点 (▲)、难点 (★)、自学 (△) | 课程目标 (√) | | | | | |
|----------------------------|--|----------|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| 第 1 章 C#概述 | (1)•NETFramework、C#语言的特点及其发展；(2)简单的三种应用程序（控制台应用程序、Windows 窗体应用程序和 Web 应用程序）的结构及其特点[了解]；(3)结合例子讲授 Visual Studio .Net 2019 中的操作方法，包括创建项目、编辑程序、生成和调试应用程序的方法[掌握]；(4)C#程序的结构、特点 (▲)。 学习重点：•NETFramework、C#语言的特点 (▲) C#程序的建立、编辑、生成和运行等基本上机操作过程 (▲) 托管程序执行的过程 (▲) | √ | | | √ | √ | |
| 第 2 章 C#程序设计基础知识 (一) | 常量和变量概念，变量的声明、初始化方法[掌握]；值类型、引用类型的差别[了解]；C#数据类型：简单类型[掌握]、枚举型 enum[了解]、结构型 struct[了解]、数据类型转换[理解]；运算符与表达式：算术运算符与表达式[掌握]、赋值运算符与表达式[理解]、关系运算符与表达式[理解]、逻辑运算符与表达式[理解]、运算符优先级[掌握]。掌握 C#的运算符和表达式的概念，理解运算符运算规则，理解表达式的使用方法，掌握各种运算符的结合性和优先级；(4) 数组和字符串：一维数组[掌握]、多维数组[掌握]、参差数组（数组型数组）[了解]、字符串(sting 类)、动态字符串类 (stringBuiler)；理解数组和字符串的概念，掌握一维数组和字符串类 string 的使用方法，了解多维数组、数组型数组的应用,了解动态字符串 stringBuilder 的用途。 相关知识点：内存及其地址分配的相关知识[了解] (△)，计算机中的数制（二进制、八进制、十六进制等）与字符编码（ASICC 码、GB2312-80 等、Unicode 码等）的相关知识[了解]，utf8 (△) | √ | | | | | |
| 第 3 章 C#程序的 流程控制 | 程序的基本结构：顺序结构（序列结构）、分支结构、循环结构、结构化程序的概念、正规程序的概念、简单正规程序的概念；分支语句：if 语句 (▲)、条件运算符 (? :)(▲)、多分支 if...else if 语句 (▲)、switch 语句 (▲)、分支语句的嵌套[理解]；理解分支的概念，掌握 if 语句和 switch 语句的使用方法 循环语句：while 语句 (▲)、do...while 语句 (▲)、for 语句 (▲)、foreach 语句 (▲) 循环嵌套[理解]；理解循环的概念，掌握 while、do/while、for、foreach 语句的使用方法，l；跳转语句：break 语句 (▲)、continue 语句 (▲)。 程序流程图符号及简单程序的流程图规定 (▲) Microsoft Visio 的操作 (△) | √ | √ | | √ | | |
| 第 4 章 面向对象程 | 面向对象基本概念：对象、事件与方法、类与对象、抽象封装继承与多态；类的定义与使用：类的声明、类的实例化、类的可访 | √ | √ | | √ | √ | |

| | | | | | | |
|------------------|--|---|---|---|---|---|
| 序设计入门 | <p>问性、值类型与引用类型；类的成员及其定义：常量与字段、属性、方法、构造函数、析构函数；方法的参数传递：按值传参、按引用传参、输出参数、引用类型的参数传递、数组型参数；方法的重载：成员方法的重载、构造函数的重载；理解面向对象的基本概念，正确分类和对象，对象的声明和对象的创建[理解]；掌握类的定义与使用方法，正确定义类的数据成员、属性和方法。[掌握]；理解类的可访问性、正确使用访问修饰符控制对类成员的访问。[掌握]；掌握类的方法的定义、调用与重载[掌握]，理解方法的参数传递的工作机制[理解]。</p> <p>理解值类型与和引用类型的区别。[理解]；理解构造函数与析构函数的作用，掌握其使用方法。[掌握]；类与对象的关系（▲），类的实现（▲），对象的创建方法（▲），通过对象名的方法调用（▲），UML 类图（▲），对象图（▲）。</p> <p>学习重点：类的定义（▲）、类的数据成员（▲）、属性和方法（▲）、构造函数（▲）方法的重载和参数传递（▲）</p> <p>方法的参数传递重点掌握参数传值、引用、输出型 3 种，理解数组型参数的传递方式</p> | | | | | |
| 第 5 章面向对象的高级程序设计 | <p>（1）静态成员与静态类：静态成员、静态构造函数、静态类；（2）类的继承性；（3）类的多态性：静态多态性与动态多态性、虚方法与动态多态性；（4）抽象类：抽象类及其抽象成员、重载抽象方法；（5）接口：接口的声明、接口的实现、接口多重继承与实现、接口的使用、抽象类与接口的比较；（6）嵌套类、分部类与命名空间。</p> <p>总体要求：掌握静态类与静态类成员的定义与使用；理解类的继承性与多态性，掌握其应用方法；理解抽象类、接口的概念，掌握它们的使用方法。</p> <p>学习重点：静态成员与静态类（▲）、类的继承性与多态性（▲）、UML 类关系图（▲）、抽象类与接口定义与使用（▲）</p> <p>难点是：虚方法的使用方法、虚方法和抽象方法的执行过程（★）继承关系中的构造函数的调用（★）、派生类对象生存期结束时的析构函数的调用顺序（★）、引入虚继承后派生类对象创建时构造函数的调用顺序（★）</p> | √ | √ | √ | √ | √ |
| 第 6 章集合、索引器与泛型 | <p>集合：集合概述、ArrayList、哈希表、栈和队列；索引器：索引器的定义与使用、索引器与属性的比较；泛型：泛型概述、泛型集合、自定义泛型</p> <p>总体要求：了解.NETFramework 类库中集合类，初步掌握常用集合的创建和操作方法。理解索引器的概念，能区别索引器与属性，掌握索引器的定义与使用。了解泛型的相关概念，初步掌握泛型接口、泛型类、泛型属性和泛型方法的使用方法。</p> <p>学习重点：集合、索引器、泛型的定义与使用</p> | √ | | | | |
| 第 7 章程序调试与异常处理 | <p>程序错误：程序错误分类、调试程序错误；程序异常处理：异常的概念、异常类、try-catch 语句、finally 语句、throw 语句与抛出异常。</p> <p>总体要求：了解程序错误的 3 种类型、熟练运用 VS2017 的调试器</p> | √ | | | √ | |

| | | | | | | | |
|---------------------------|---|---|---|---|---|---|--|
| | <p>调试程序错误、了解异常和异常处理的概念、学会使用 try-catch-finally 及 throw 语句来捕获和处理异常。异常处理过程 (▲), 自定义异常类的设计 (▲)。</p> <p>学习重点: 调试程序错误的方法 (▲)、try-catch-finally 结构及其使用方法 (▲)</p> <p>学习难点: 自定义异常类的设计及抛出自定义异常对象 (★) 再次抛出异常的处理 (★)</p> | | | | | | |
| 第 8 章 基于事件驱动的程序设计技术 | <p>基于事件的编程思想; 委托: 委托概述、委托的声明、实例化与使用、委托与匿名函数、多路广播与委托的组合; 事件: 声明事件、订阅事件、触发事件; 基于事件的 Windows 编程: Windows 应用程序概述、Windows 窗体与事件驱动编程</p> <p>总体要求: 事件源、侦听器、事件处理程序的概念[了解]、委托的声明、实例化和使用方法[掌握]、多路广播及其应用[了解]、事件的声明、预定和引用[掌握]、Windows 窗体和控件的常用事件[了解]、事件和事件方法之间的关系[理解]。</p> <p>学习重点: 委托的声明、实例化和使用方法 (▲)、事件的声明、预定和引用</p> | √ | | | | | |
| 第 9 章 多线程和异步编程 | <p>多线程的概念: 线程和进程、线程的生命周期和状态、线程的优先级; 线程创建和控制: 创建和启动线程、控制线程</p> <p>多线程的同步: 线程安全、线程同步策略; 线程池: 线程池管理、ThreadPool 类; 异步编程: 异步编程和多线程、异步编程模式。</p> <p>总体要求: 线程和进程的概念[了解]、使用 C#进行多线程的创建及简单控制[掌握]、线程的同步策略[掌握]、线程池技术[了解]、异步编程的设计与实现技术 [掌握]。</p> <p>学习重点: 线程的启动、管理和终止 (▲)、线程安全的实现 (▲)、异步编程的实现方法 (▲)。</p> | √ | | | | | |
| 第 10 章 Windows 程序的界面设计 | <p>窗体与控件概述: Windows 窗体、窗体的控件; 常用输入与输出控件: 文本显示控件、文本编辑控件、按钮控件、图像显示控件; 列表与选择类控件: 选项与选项组、列表类控件、其他选择类控件、日历与计时器控件对话框及其选项卡设计: 对话框概述、对话框的选项卡、消息框、通用对话框; 菜单、工具栏和状态栏: 菜单、工具栏、状态栏; SDI 和 MDI 应用程序: 创建 SDI 应用程序、创建 MDI 应用程序。</p> <p>(▲) 菜单的编辑以及事件关联过程、模态窗体和非模态窗体 (▲) 添加对话框资源的步骤 (▲), 利用 Visual studio 资源编辑器设计窗体界面 (▲), 在自定义窗体与主窗体之间传送数据 (★) 多文档界面 MDI 中母窗体与子窗体之间的导航 (★)</p> | √ | √ | √ | √ | | |
| 第 11 章 C#数据库编程技术 | <p>ADO.NET 概述: ADO.NET 的架构、ADO.NET 的一般使用步骤; 使用 ADO.NET 访问数据库: 使用 Connection 连接数据库、使用 Command 对象访问数据库、使用 DataReader 对象访问数据库、使用 DataAdapter 与 DataSet 对象操作数据库; LINQ to SQL 编程: LINQ 概述、LINQ 的查询子句与表达式、LINQ to SQL 的应用</p> <p>总体要求: 掌握 ADO.NET 的使用方法[掌握]、包括 Connection、Command、DataSet、DataAdapter、DataReader、DataGridView 控</p> | √ | | | √ | √ | |

| | | | | | | |
|----------------------------------|--|---|---|---|---|--|
| | <p>件的使用方法;LINQ 基本语法[了解]、LINQ to SQL 的使用方法[掌握]。</p> <p>学习重点: 使用 Connection 和 Command 对数据库的操作 (▲)、掌握 DataSet、DataAdapter、DataReader 对象的作用和使用方法 (▲)、使用 DataGridView 结合 ADO.NET 对数据库的修改和查询操作 (▲)</p> | | | | | |
| 第 12 章 文件操作与 编程技术 | <p>文件的输入输出: 文件 I/O 与流、读写文本文件、读写二进制文件、对象的序列化; 文件操作控件: SaveFileDialog OpenFileDialog 控件、FolderBrowserDialog 控件、简易写字板程序。</p> <p>总体要求: 文件与流的区别[理解], 常用的操作流的类的功能[了解]、文本文件或二进制文件读写方法[掌握]、序列化和反序列化的概念[了解]、序列化和反序列化的实现方法[掌握]、利用这些控件来打开或保存文件的实现方法[掌握]。</p> <p>学习重点: 文本文件或二进制文件读写 (▲)、文件操作控件使用 (▲)。XML 文档编程: XML 概述、XML 文档的创建、XML 文档的查询、XML 文档的编辑。</p> <p>总体要求: XML 的概念和及其基本的语法规则; [了解]</p> <p>访问 XML 的相关技术及其特点(包括 DOM、XPath 和 XQuery 等)[了解], 操作 XML 文档的编程方法, 包括创建 XML 文档、查询和编辑 XML 数据等[掌握], LINQ 的相关概念)[了解], LINQ 查询的语法规则[掌握], LINQ to XML 和这两种技术的应用方法[掌握], DOM 技术在 XML 中的应用 (△)</p> <p>LINQ 查询 (△)、LINQ to XML (△) 和 LINQ to SQL (△) XML 文档的创建 (▲)、XML 文档的查询 (▲)、XML 文档的编辑 (▲)</p> | √ | | | | |
| 第 13 章 网 络应用与面 向服务程序 设计 | <p>网络编程基础: 计算机网络概述、计算机网络的通信协议、System.Net 概述; Socket 编程: Socket 编程概述、TCP 应用编程、UDP 应用编程; FTP 与文件传输编程: WebRequest 类和 WebResponse 类、FTP 客户端的实现; 基于 Web API 的面向服务编程: ASP.NET Web API 概述、Web API 服务器端编程、HttpClient 客户端编程。</p> <p>总体要求: 熟悉 System.Net 及其子命名空间中的常用类、熟悉 Socket、TcpListener、TcpClient 和 UdpClient 类, 掌握它们的编程与应用方法。熟悉 WebRequest 和 WebResponse 类、学会 FTP 客户端的编程方法; 了解 Web API 框架, 学会基于 Web API 的 Web 服务的定义与使用方法</p> <p>学习重点: 掌握 System.Net 及其子命名空间中常用类的使用方法 (▲)、掌握有关 Socket、TcpListener、TcpClient 和 UdpClient 的编程方法(▲); 掌握基于 Web API 的 Web 服务的定义与引用(▲)</p> | √ | √ | √ | √ | |
| 第 14 章 多媒体编程 技术 | <p>GDI+绘图: GDI+概述、创建 Graphics 对象、颜料、钢笔和画刷、线条和图形的绘制、图像和文本的绘制、坐标系统及变换</p> <p>Windows Media Player 组件的使用:</p> <p>Windows Media Player 组件的介绍、Windows Media Player 组件的使用</p> <p>总体要求: 了解 GDI+的组成和工作机制, 了解 System.Drawing 命</p> | √ | | | | |

| | | | | | | | |
|------------|---|--|--|--|--|---|---|
| | <p>名空间；理解画面类 Graphics、钢笔类 Pen、画笔类 Brush 和颜料类 Color 的关系，掌握创建 Graphics、Pen、Brush 对象的方法；学会绘制各种图形的方法（包括点、线条、曲线、弧线、折线、矩形、椭圆、多边形等）；掌握图像及文本的呈现方法。了解 Windows Media Player 组件对象模型，掌握其使用方法。</p> <p>学习重点：GDI+的应用（▲）、Windows Media Player 组件的使用</p> | | | | | | |
| 自主选题小组综合设计 | <p>课程综合设计（自主选择适度规模的软件设计题目，综合运用所学知识，完成软件界面设计、功能及结构分析、算法设计、代码设计、总结报告撰写，2-3 人合作完成）</p> | | | | | √ | √ |

[掌握]：指学生能根据不同情况对某些概念、定律、原理、方法等在正确理解的基础上结合事例加以运用，包括分析和综合。

[理解]：指学生能用自己的语言叙述、解释、归纳，并能把某一事实或概念分解为若干部分，指出它们之间的内在联系或与其他事物的相互关系。

[了解]：指学生应该辨认的科学事实、概念、原则和术语等，知道事物的分类、过程及变化倾向，包括必要的记忆。

[Δ]：指学生自学或教师粗讲。

2、支撑毕业能力项的教学内容

毕业能力[1] 工程知识：面向对象编程基础、面向对象设计方法、数据库编程基础、文件和流编程基础

毕业能力[3] 设计/开发解决方案：教学内容中的教务管理案例教学、Windows GUI 界面设计及程序设计、Web 页面设计与后台数据库设计及代码设计、小组综合设计

毕业能力[5]使用现代工具：Visual Studio 2017（及以上）的熟悉操作，Microsoft Visio 流程图绘制工具，Microsoft Access2017 简单数据库的创建、表的建立、数据录入、SQL 语句使用

四、教学环节安排及要求

本课程共 48 学时，其中理论教学：32 学时，实验教学：16 学时，课外安排小组综合设计（学生均约需投入 10 天时间，每天约 2 小时）

理论教学：重点讲清楚 C#语言面向对象程序设计基础、Windows 窗体应用程序设计基础、Web 程序设计基础，约 28 学时，以讲授为主，辅以课外书面作业，学生课外预习，安排约 4 小时的习题评讲活动。

书面作业按学生专业类别要求如下：

非电子信息类专业：第 1 章：题 1、题 2；第 2 章：题 1、题 2；第 3 章：题 1、题 2、题 3.1、题 3.2 题 3.3、第 4 章：题 1、题 2、题 3；第 5 章：题 1、题 2；第 6 章：题 1、题 2；第 7 章：题 1、题 2；第 8 章：题 1、题 2；第 10 章：题 1、题 2；第 12 章：题 1、题 2；第 14 章：题 1、题 2；

电子信息类专业：第 1 章：题 1、题 2；第 2 章：题 1、题 2；第 3 章：题 1、题 2、题 3.4、题 3.5、题 3.6、第 4 章：题 1、题 2、题 3；第 5 章：题 1、题 2；第 6 章：题 1、题 2；第 7 章：题 1、题 2；第 8 章：题 1、题 2；第 9 章：题 1、题 2；第 10 章：题 1、题 2；第 11 章：题 1、题 2；第 12 章：题 1、题 2；第 13 章：题 1、题 2；第 14 章：题 1、

题 2:

实验教学: 本课程针对主要知识点设计了难度不同的配套实验, 由任课教师在实验课上指导完成。具体实验教学内容与教学要求见表 3, 每一教学年度将更新调整实验内容, 使学生能仅依赖上一轮学员的报告资料就能完成上机编程训练的任务。

表 3 课内实验教学内容与要求

| 序号 | 实验项目名称 | 实验主要内容 | 实验类型 | 计划学时 | 开出要求 | 每组人数 |
|----|----------------|--|------|------|------|------|
| 11 | 数据类型和基本语句语法的运用 | 编写一个控制台程序, 对于输入的两个正实数, 代表的长方形的面积和周长, 计算出该长方形的边长。 编写一个 C#窗体应用程序, 对于输入的正整数 n, 然后计算 $1!+2!+3!+\dots+n!$ 的值并输出。 | 验证设计 | 4 | 必做 | 1 |
| 22 | 类的运用 | 定义一个大整数类, 使得基于此类可以实现对最高有 100 位的整数进行加、减法运算, 以及进行大整数比较 (包括相等、大于和小于的比较) | 验证设计 | 4 | 必做 | 1 |
| 33 | 基本异常处理 | 编写一个能够进行加、减、乘、除的计算器程序 (窗体应用程序), 并能够处理可能产生的异常。 | 验证 | 2 | 必做 | 1 |
| 44 | 窗体应用程序开发方法 | 编写一个包含两个 ListBox 控件的窗体应用程序, 要求实现下列功能: (1) 通过鼠标可以将一个 ListBox 控件中的任意一项拖到另一个 ListBox 控件中, 反之亦然; (2) 拖动时, 鼠标呈手势状, 使得拖动操作更具形象化。 | 验证设计 | 4 | 必做 | 2 |
| 55 | 文本文件读写方法 | 利用文本文件的读写方法, 开发一个“记事本”程序, 要求能够实现文件的打开、编辑、保存功能, 并能够处理可能出现的异常。 | 验证 | 2 | 必做 | 1 |
| 66 | 数据库应用程序开发 | 创建一个数据库应用程序, 使它能够对表 teacher 进行简单的数据浏览、插入、更新和删除操作。 | 验证设计 | 4 | 必做 | 1 |
| 77 | Web 数据库应用程序开发 | 创建一个 Web 服务程序, 它针对指定的数据表提供共享的 DataSet 对象; 同时创建一个窗体应用程序, 它通过利用由 Web 服务共享的 DataSet 对象来实现数据浏览功能。 | 验证设计 | 4 | 必做 | 2 |

注: 指导教师备课时提前准备好实验参考案例的运行、阅读理解以及改进性设计的要求, 共提供了总共 24 个学时的实验列表, 可根据需要从中选择 16 学时的实验方案。

五、教授方法与学习方法

教授方法:

1、课堂讲授是知识传授的主要方式。在课堂教学中, 采用 PPT 展示并讲解的方式进行教学, 对于部分案例涉及到数学原理, 需用粉笔在黑板上进行推导和说明; 根据需要还进行专题学习和讨论、课堂即时测验等, 就重点、难点问题进行分析和学习; 此外, 在讲授 Windows 应用程序设计案例时, 需要在笔记本电脑上配置好运行环境, 根据实际需要, 通过实时操作并投屏等方法进行演示, 还通过视频、动画等方式进行展示。总之, 通过多

种方式，积极调动学生的好奇心和积极性，进行引导，以传授 C#、.Net 编程知识，讲授新内容前需简短回顾前一周次理论课堂学习的知识点，注意同时强调知识点的衔接、知识结构的贯通，在传授知识的同时，注意培养学生的批判性思维，对于在案例讲解中提取思想政治元素，培养学生的精神情操。

第 9 章、第 12 章、第 13 章、第 14 章采取课外自学加在线答疑的方式教学。综合设计任务采用小组课外合作完成方式，教师提供必要的技术支持和选题难度程度、设计方案的合理性等的方面的把控。

2、实验教学

实验教学是 C#编程技能训练的一种主要教学环节，主要包括验证性实验和设计性实验两种，验证性实验可根据学生的硬件资源条件情况安排在课外完成，设计性实验在机房集中指导。

3、作业练习

作业练习是对知识重复训练的一种能力培养方式。任课教师根据教学进度安排适量的作业，此外还安排问题解答时间、安排习题课等，点评普遍性与重要性的问题。

4、综合设计训练

针对重要的问题或涉及的复杂应用系统的工程问题，教师根据学生掌握的知识和能力情况适当安排综合性设计型大作业，通过分组讨论、答辩、提交报告等方式帮助学生培养解决复杂工程问题的能力。

学习方法：本课程要求学生既扎实学习理论知识，又要动手实践训练才能有收获。课前预习下一周拟进行的学习任务，课上认真听讲，课后及时完成书面作业并修改前一次的错误解答，实验前认真阅读实验指导书文档，理解实验任务，弄清楚设计题的具体要求，按操作流程指导先跑通参考样例程序，阅读理解其原理逻辑，再思考如何给出改进性设计要求完成程序设计并调试成功后，还需要给出必要的测试数据，再编撰实验报告，书写报告时需写出分析解决设计问题的思路，讨论设计结果，对于计算性复杂的任务，要分析所采用的算法原理，学会比较解决同一问题的不同算法的时间和空间效率。

可以利用网络上的同类课程或学习视频来辅助本课堂学习任务的完成，但需要注意避免发散性被动学习，或者过分迷恋最新版本开发环境的新增特性而忽视基础的稳固的专业知识的学习与理解。

综合设计任务的完成要学会利用各种文献资源，提炼出适合自身任务的可用成分，把表示工程问题的信息模型设计成最简洁的最不冗余的形式，构思程序的模块结构时要注意模块具有高内聚集性，不同模块之间相关性低的特点，设计操作界面时需考虑操作者的方便性。

六、学时分配

表 4 各章节学时分配表

| 章节 | 主要内容 | 学 时 分 配 | | | | | 合计 |
|--------|---|---------|--------|--------|--------|--------|-------|
| | | 讲 授 | 习 题 | 实 验 | 讨 论 | 其 它 | |
| 第 1 章 | .NET 架构概述 C#概述、3 种应用程序基本结构 | 1.5 | 0.10 | 1 | 0 | 0 | 2.6 |
| 第 2 章 | 变量与常量、数据类型及转换、运算符与表达式、 数组、字符串 | 3.5 | 0.25 | 0.75 | 0 | 0 | 4.5 |
| 第 3 章 | 流程控制语句、控制台程序、简单窗体程序的结 构分析 | 6 | 0.65 | 3.25 | 0 | 0 | 9.90 |
| 第 4 章 | 面向对象程序设计（一）：类、常量与字段、成员 可见性、属性（只读、只写、可读可写）、方法（函 数）、方法的参数传递、构造函数、析构函数、方 法的重载、构造函数的重载 | 6 | 0.5 | 4 | 0 | 0 | 10.50 |
| 第 5 章 | 面向对象程序设计（二）：继承、派生类成员对基 类成员的可访问性、派生类构造函数的设计及执 行过程、派生类析构函数的执行过程、虚方法及 动态多态性、抽象方法与抽象类、静态成员及静 态类、嵌套类（/内部类）、分部类与命名空间、接 口 | 3.5 | 0.20 | 2 | 0 | 0 | 5.70 |
| 第 6 章 | 集合、索引器与泛型 | 0.5 | 0.10 | 0 | 0 | 0 | 0.6 |
| 第 7 章 | 程序调试与异常处理 | 1.25 | | 0.5 | 0 | 0 | 1.75 |
| 第 8 章 | 基于事件驱动的程序设计技术、委托、.NET 泛型 事件委托类、Windows 事件驱动程序运行过程、 事件的订阅与点火 | 1.75 | 0.10 | 0.5 | 0 | 0 | 2.35 |
| 第 9 章 | 多线程和异步编程（课外自学） | 0 | 0 | 0 | 0 | 0 | 0 |
| 第 10 章 | Windows 程序的界面设计、多窗体程序设计 | 1.5 | 0.10 | 3 | 0 | 0 | 4.60 |
| 第 11 章 | C#数据库编程技术（ADO.NET 的结构及有关对象 的使用）（电子信息类学生课外执行） | 0 | 0 | 0 | 0 | 0 | 0 |
| 第 12 章 | 文件操作与编程技术（文件、流、XML 基础） | 1.5 | 0 | 0.5 | 0 | 0 | 2 |
| 第 13 章 | 网络应用与面向服务程序设计（电子信息类学生 课外执行） | 0 | 0 | 0 | 0 | 0 | 0 |
| 第 14 章 | 多媒体编程技术（只讲 GDI+基本部分，掌握简单 图形绘制方法） | 1 | | 0.5 | | | 1.5 |
| 综合设计 | 根据学生兴趣，自主选题结合教师拟定设计任务 选择任务，3-4 人分小组进行（课外执行） | 0 | 0 | 0 | 0 | 0 | 0 |
| 期末考试 | | 2 | | | | | 2 |
| 合计 | 32 | 30 | 2 | 16 | | | 48 |

七、考核与成绩评定

鉴于程序设计是一种理论性实践性并重的生产方式，本课程按照是一个理论性实践性并重的“讲一、练二、考三”的教学理念，在教学过程中既注重基本原理的教学，又加强实

践训练，训练分上机实验和自主综合设计 2 个环节，上机实验配合课堂理论教学，旨在巩固理论知识的理解和应用，实验内容涉及的问题规模交小，所需知识点具体明确，而且有参考解决方案，学生需要理解的基础上做适当改进，对学生学习独立知识单元及其应用方法起到巩固和加强映像的作用，安排在公共机房内由实验指导教师指导学生完成；综合设计训练安排在临近期末的课外，由学生自主选题按小组合作完成，期末需提交设计报告文档并答辩。

期末考核采取平时成绩（10%）+实验成绩（20%）+综合设计（10%）+期末考试（基本理论知识考试，60%）的方式进行。

平时成绩包括随堂练习及出勤率（5%），采取课堂签到登记以及不定期随机课堂练习或问答方式开展，出勤次数低于 8 次者给 0 分，出勤次数 8-10 次给 2 分，出勤次数 10 次以上 15 次一下但随堂练习回答质量一般给 3 分，出勤次数在 10 次以上随堂练习质量较好给 4 分，出勤次数 15 次但随堂练习回答质量一般给 4 分，出勤次数 14 次和 15 次但随堂练习回答质量一般给 4 分，出勤次数 12 次以上随堂练习回答质量较好给 5 分；课后作业完成质量（5%），根据按进度完成课外纸质作业或文献检索任务的数量和质量评价，按进度全额完成，作业质量等级 A 数量占比高则给 5 分，按进度完成作业质量等级 B 占次数高则给 4 分，按进度完成但作业质量等级 C 占次数多则给 4 分，不按进度完成但补齐作业给 3 分，缺交部分作业给 2 分，其他给 0 分。

实验成绩评定主要根据实验现场表现、实验报告质量评价，报告内容完整，思路分析清楚，流程图、类图、类关系图规范而且与程序对应、正确，有设计结果及分析改进设想，按期提交分值较高，期末补交则得分较低，具体计算规则实验时由指导教师宣布。

综合设计成绩根据答辩情况评定，要求设计内容具有一定的综合性，有自主创新点或对原作品有改进，报告内容完整，设计思路叙述清楚，数学原理模型、流程图、类图、类关系图规范而且与程序对应、正确，有设计结果及分析改进设想者较优秀，分为优秀、良好、中等、合格，分别给 9-10 分，8 分，7 分，6 分，没有设计成果者 0 分。

期末考试可采取闭卷考试方式，任课教师在期末结课前 2 周给出复习思考题及参考解答，考试题型及分值分布明确公布，试题考核内容涉及 C#语言基础知识（运算符优先级与表达式、流程控制语句、数组、字符串）、面向对象编程基础、控制台程序的输入输出语句、委托类型与事件驱动方式、Windows 窗体程序设计基础、Windows 界面设计基础等，期末成绩占学期总评的比重为 60%。

也可采取开卷方式，在采取开卷考试方式的学期，任课教师需在期末结课前二周向学生公布，并将期末题目类型及分值比重明确公布，试题考核内容涉及的知识点与闭卷方式相同，但应增加设计题目的灵活度并适当增加设计题目的分值。

表 5 考核方式及成绩评定分布表

| 考核方式 | 比例 (%) | 主要考核内容 |
|----------|--------|--|
| 作业 | 5 | 按教学进度完成的课外书面作业的数量和质量 |
| 随堂练习及出勤率 | 5 | 回答问题的质量与表述能力，出勤 |
| 实验 | 20 | 按期提交的报告数量和报告撰写质量（目的明确、步骤完整、有分析设计思路、流程图、类图、类关系图规范） |
| 综合设计 | 10 | 选题的难易程度、设计的完整性、思路分析清晰度、流程图和类图是否规范 |
| 期末 | 60 | C#语言基础知识（数据类型、运算符与表达式、流程控制语句、数值与字符串之间相互转换）、面向对象程序设计基础、Windows 程序编程基础、委托与事件的发布、订阅 |

制定者：李家军

批准者：张利国

2020 年 7 月