

安徽信息工程学院

《面向对象程序设计》课程设计与实施方案

一、课程基本信息

1. 课程编号：CSE2005
2. 课程类别/属性：专业核心课（必修）
3. 学时/学分：80 学时（16/32，24/8）/3 学分
4. 适用专业：软件工程、网络工程、计算机与科学技术
5. 面向群体：计算机与软件工程学院大一新生
6. 开课学期：春季学期
7. 前置课程：《计算思维导论（C 语言）》
8. 后续课程：《软件建模基础实践》《数据结构与算法》
9. 教材：雷大正，王啸楠，丁德成等著. 面向对象程序设计 C++实现. 北京：机械工业出版社，2018.03

二、课程设计理念与思路

《面向对象程序设计》课程设计遵循 ADDIE 教学系统设计框架，如图 1 所示。

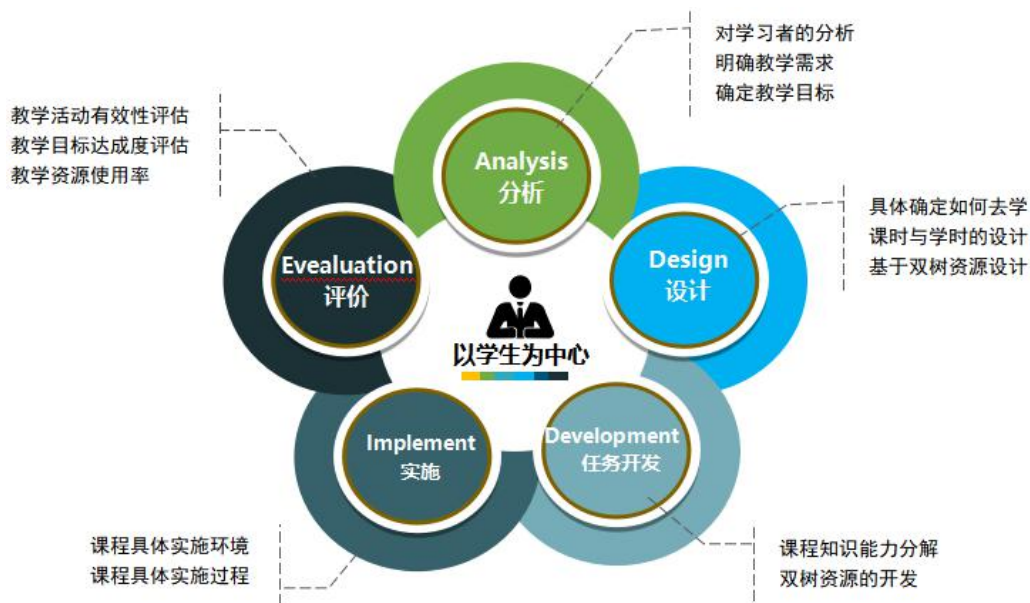


图 1 《面向对象程序设计》课程设计理念与思路图

整个模型设计分为 5 个组成部分，分别是分析 (Analysis)、设计 (Design)、任务开发 (Development)、实施 (Implement)、评价 (Evaluation)。将“以学生为中心”作为整个教学设计的核心理念，主要包含三个方面的内容，即学生要学什么 (学习目标的制定)、学生如何去学 (学习策略的应用)、如何去判断学生已达到学习效果 (学习考评实施)。

三、课程目标 (Analysis)

本课程培养目标基于三个培养维度进行设置，分别是专业知识、工程能力和职业素养。按照“以能力为本位、以编程实践为主线”的总体设计要求，主要以培养学生面向对象思维为重点，采用软件工程模块化、能力导向及基于“解决问题的知识架构模式”的思路重新构建课程知识体系，以案例、任务为驱动模式，结合探究式模式构建核心学习内容，要求学生不仅要掌握 C++ 程序设计的基本原理及编程技术，而且应掌握至少一个 C++ 语言开发环境的使用及 C++ 程序调试和测试的方法，并通过团队综合项目实现，培养学生团队合作意识以及沟通交流能力，为后续的专业课程奠定程序设计基础，支撑专业学习成果中相应指标点的达成。

课程目标对学生能力要求如下：

课程目标 1. 了解并掌握面向对象思维方式，熟练掌握 C++ 语法，能够将抽象建模后的项目功能通过编程的方式实现其具体的逻辑关系，解决复杂工程问题。

课程目标 2. 掌握面向对象思维分析方法，面向软件工程领域复杂问题能够合理进行抽象和分解，并能够用于解决复杂工程问题。

课程目标 3. 培养实际建模能力，能够设计针对复杂软件工程问题的解决方案，设计满足特定需求的软件系统。

课程目标 4. 熟练掌握 VS 编程工具对计算机复杂工程问题进行模拟、仿真和预测。

课程目标 5. 具备综合项目实践能力，熟练掌握 VS 编程工具以及 C++ 语言开发环境，并能够理解其局限性。

课程目标 6. 具备团队合作意识以及沟通交流能力，能够在团队中承担个体、团队成员以及负责人的角色。

表 1 课程目标对毕业要求的支撑关系表

毕业要求	毕业要求分解指标点	课程目标	培养目标点
1.工程知识	1-4: 掌握软件工程专业核心知识, 并能够用于解决复杂工程问题。	课程目标 1	专业知识
2.问题分析	2-1: 能够运用数学和自然科学知识识别和表述复杂工程问题。	课程目标 2	专业知识
3.设计/开发解决方案	3-1: 能够在计算机软硬件开发项目中进行系统概要设计和详细设计。	课程目标 3	工程能力
5.使用现代工具	5-2: 能够开发、选择和使用恰当的技术和工具对计算机复杂工程问题进行模拟、仿真和预测。 5-3: 能够针对计算机软硬件开发的需要选择和使用适当的平台环境和开发工具, 并能够理解其局限性。	课程目标 4 课程目标 5	工程能力
10.个人和团队	10-3: 能够理解个人在团队中的角色划分, 且胜任相应的角色职责。	课程目标 6	职业素养

四、课程教学设计思路 (Design)

整门课程共 80 学时, 分为线下课时 (48 课时) 以及线上学时 (32 学时)。其中线下课时分为理论课时 (16 课时) 和上机课时 (32 课时), 共计 24 讲; 线上学时分为主题案例自主学习学时 (24 学时) 和项目实践自主练习学时 (8 课时)。

结合课程教学目标, 将目标点从大至小逐步划分, 明确每一个教学目标在整个教学目标中的支撑点, 即主题知识树下的每一个主题案例, 明确每个支撑点能够让学生达到什么能力, 最终形成的综合能力需要符合整门课程的教学目标。最终落实到每一个课时与学时的教学安排设计中。

《面向对象程序设计》课程教学总设计如表 2 所示。

表 2 《面向对象程序设计》课程教学总设计表

教学内容	教学安排	线下课时		线上学时		课堂教学形式与手段	教学活动设计		
		理论	上机	主题	项目		课前: 知识前置学习	课中: 知识强化实践	课后: 知识巩固拓展
主题知识单元与综合项目	共 24 讲	16	32	24	8				
0.课程导入				1		自主学习	观看主题案例 0 视频与课件		
1.面向对象思维建立	第 1 讲	1				讲授+案例分析+交流	观看主题案例 1 视频与课件	1. 通过讲授现实真实世界案例的分析, 初步了解类与对象之间的关系。	
2.生活中的类与对象			1						1.完成测试案例 2 练习并提交分析文档

								2. 通过对案例分析讨论, 强化学生类与对象的概念	
3.类之间的关系	第 2 讲	1	1	1		讲授+案例分析+交流	观看主题 3 案例视频与课件	1. 讲授类之间五种关系 2. 通过对案例分析讨论, 掌握类关系在实际场景中的体现	1. 完成测试案例 3 练习并提交分析文档 2. 完成在线测试
4.UML 类图	第 3 讲	1	1	1		讲授+案例分析与设计	观看主题 4 案例视频与课件	1. 讲授类图的基本组成以及各个组件含义 2. 通过对案例分析与设计实现一个完整的类图设计	1.完成测试案例 4 练习并提交类图设计图
5.UML 时序图				1		自主学习与考核	观看主题 5 案例视频与课件		1.完成测试案例 5 练习并提交时序图设计图
6.C++结构程序设计	第 4 讲	1	1	1		讲授+案例分析与实现		1. 讲授 C++语法与 C 语法不同之处 2. 通过案例分析完成场景案例的代码实现, 强化知识点	1.观看主题 6 案例视频与课件并完成测试案例 6 的代码实现
7.C++函数				1		自主学习与考核	观看主题 7 案例视频与课件		1.完成在线测试
8.C++数组	第 5 讲	1	1	1		讲授+案例分析与实现	观看主题 8 案例视频与课件	1.完成一维数组的数据增删改查案例分析与实现	
9.C++容器	第 6 讲		2	1		讲授+案例分析与实现	观看主题 9 案例视频与课件	1.完成容器的数据增删改查案例分析与实现	1.采用数组或容器方法实现一组集合数据的增删改查案例代码实现
10.C++指针	第 7 讲		1	1		讲授+案例分析与实现	观看主题 11 案例视频与课件	1. 讲解指针与引用的不同存储方式 2. 讲解引用的定义与初始化方法 3. 通过针对错误代码调试, 完成异常处理的讲解	1.完成测试案例 11 的错误代码修复, 并提交正确代码
11.C++程序调试与异常处理			1	1					
12.C++类与对象	第 8 讲 第 9 讲	1	3	1		场景模拟+案例实现	观看主题 12 案例视频与课件	1. 通过场景模拟完成类的抽象与对象实现 2. 学生分组完成测试案例 12 代码实现	1. 完成在线测试 2. 完成测试案例 12 的代码实现并提交源代码

13.C++类的依赖关系	第 10 讲	1	1	1		场景模拟+案例实现	观看主题 13 案例视频与课件	1. 通过场景模拟完成类依赖关系实现 2. 学生分组完成测试案例 13 代码实现	1.完成测试案例 13 的代码实现并提交源代码
14.C++类的关联关系				1		自主学习与考核	观看主题 14 案例视频与课件		1.完成测试案例 14 的代码实现并提交源代码
15.C++类的组合关系				1		自主学习与考核	观看主题 15 案例视频与课件		1.完成测试案例 15 的代码实现并提交源代码
16.C++类的聚合关系	第 11 讲	1	1	1		场景模拟+案例实现	观看主题 16 案例视频与课件	1.通过场景模拟完成类依赖关系实现 2.学生分组完成测试案例 16 代码实现	1.完成测试案例 16 的代码实现并提交源代码
17.C++类的继承关系				1		自主学习与考核	观看主题 17 案例视频与课件		1.完成测试案例 17 的代码实现并提交源代码
18.C++成员函数	第 12 讲	1	1	1		讲授+案例分析与实现	观看主题 18 案例视频与课件	1. 讲授成员函数的概念以及两种实现方法 2. 通过案例分析与实现完成成员函数功能实现	1.完成测试案例 18 的代码实现并提交源代码
19.C++构造与析构	第 13 讲 第 14 讲	1	3	1		讲授+案例分析与实现	观看主题 19 案例视频与课件	1. 讲授构造函数和析构函数的概念与特点 2. 通过案例分析与实现完成构造函数与析构函数的调用	1. 完成在线测试 2. 完成测试案例 19 的代码实现并提交源代码
课程综合项目 V1.0	第 15 讲	1	1		2	项目交流+辅导答疑	观看课程综合项目 V1.0 教学视频	1.带学生梳理项目框架以及设计思路 2.学生分组进行讨论交流,并根据教学视频进行源代码实现	1.完成课程综合项目 V1.0 代码实现并提交源代码(通过注释语句表述清楚所有功能模块)
20.C++封装 private 访问权限	第 16 讲	1	1	1		讲授+案例分析与实现	观看主题 20 案例视频与课件	1. 讲解封装在面向对象思维中的意义,并通过 private 访问权限实现场景案例 2. 讲解 get 与 set 函数使用场景与技巧	1. 完成在线测试 2. 完成测试案例 20 的源代码实现并提交
21.C++友元函数与静态成员				1		自主学习与考核	观看主题 案例 21 视频与课件		1.完成测试案例 21 的源代码实现并提交

22.C++单继承	第 17 讲	1	1			场景模拟+案例实现		1.通过场景模拟完成单继承实现 2.学生分组完成测试案例 22 代码实现	1.观看主题案例 22 的视频与课件 2.完成测试案例 22 的源代码实现并提交
23.C++继承中的构造与析构	第 18 讲 第 19 讲	1	3	1		讲授+案例分析与实现+随堂测试	观看主题 23 案例视频与课件	1. 讲授继承关系中的构造与析构函数的调用规则及顺序 2. 通过案例分析与实现突出构造函数与析构函数在继承关系中的作用 3. 通过随堂测试考核学生知识点掌握情况	1.完成在线测试
24.C++静态多态				1		自主学习与考核	观看主题 24 案例视频与课件		1.完成在线测试
25.C++动态多态	第 20 讲 第 21 讲	1	3	1		场景模拟+案例实现+随堂测试	观看主题 25 案例视频与课件	1. 通过案例分析与实现讲授虚函数的实现方法以及存在意义 2. 学生分组完成测试案例 25 代码实现 3. 通过随堂测试考核学生知识点掌握情况	1.完成测试案例 25 的源代码实现并提交
26.C++文件与流				1		自主学习与考核	观看主题 26 案例视频与课件		1.完成测试案例 26 的源代码实现并提交
课程综合项目 V2.0	第 22 讲	1	1		2	项目交流+辅导答疑	观看课程综合项目 V2.0 教学视频	1.带学生梳理项目框架以及设计思路 2.学生分组进行讨论交流,并根据教学视频进行源代码实现	1.完成课程综合项目 V2.0 代码实现并提交源代码(通过注释语句表述清楚所有功能模块)
综合评价项目辅导答疑课一	第 23 讲		2		2	项目交流+辅导答疑	1.完成分组 2.完成团队任务分配	1.学生以组为单位梳理项目框架以及设计思路 2.学生根据分组任务实现项目代码 3.讲师辅导答疑	1.学生继续完成项目代码开发
综合评价项目辅导答疑课二	第 24 讲		2		2	项目交流+辅导答疑		1.学生根据分组任务实现项目代码	1.完成整个项目代码开发,并提交源代码

								3.讲师辅导答疑	2.完成项目答辩材料编写,包括答辩 PPT,项目运行讲解视频等
--	--	--	--	--	--	--	--	----------	---------------------------------

五、课程知识能力分解与双树资源映射关系 (Development)

《面向对象程序设计》课程共涉及 3 个课程综合项目，每个项目共计 3 个版本（版本 1-2 为核心必修项目，版本 3 为扩展选修项目），涉及主题知识单元 30 个（0-26 为核心必修主题，27-29 为扩展选修主题），共涵盖课程知识点 97 个，知识点覆盖率为 100%。其中课程综合项目版本 1 功能涵盖主题知识单元 1-19，综合项目版本 2 功能涵盖主题知识单元 20-26，综合项目版本 3 功能涵盖主题知识单元 27-29，版本功能之间呈递进关系。

整门课程知识点采用 Bloom 分类法，从大至小逐步划分，明确每一个知识点的教学目标在整个教学目标中的支撑点，每个支撑点能够让学生达到什么能力，最终形成的综合能力需要符合整门课程的教学目标。

考虑到 Bloom 分类法的复杂度，为简化学生对知识点的理解，仅采用 3 个学识层级，Bloom 属性分别使用字母 K、C 或 A 来表示教育要求，这些字母所代表的意义如表 3 所示。课程知识能力分解与双树资源映射关系如表 4 所示。

表 3 Bloom 属性对应关系表

属性	对应关系
知识 (K)	要记住学习过的内容，要求对信息的观察和记忆能力，即“记得有价值的信息”（如：日期、事件、位置、主要概念、主要内容）。
理解 (C)	明白知识表示的信息和含义。（如：能将知识转化到新的上下文环境、能解释事实、比较、对比、排序、分组、推断因果关系和预测结果等）。
应用 (A)	新的、具体环境下，应用所学知识的能力。（如：使用信息、方法、概念和原理解决需要相关技能或知识的问题）。

表 4 课程知识能力分解与双树资源映射关系表

序号	课程综合项目	主题知识单元	课程知识点 (K/C/A)	目标考核(能力点)
1		0.课程导入	课程教学目标	了解课程在整个专业学习过程中的定位与培养目标，以及该门课程在后期实际工作中的企业定位与就业方向与就业前景
2			课程专业定位	
3			课程企业定位	
4			课程企业就业方向与就业前景	
5	综合项目版本 1 (LostCraft 项目)	1.面向对象思维建立	面向对象思维概念【K】	了解面向对象概念与特性，区分面向对象过程与面向对象
6	(三子棋项目)		面向对象思维方式【C】	掌握面向对象思维方式与开发思路
7	(文字闯关项目)		通过面向对象思维方式实现场	熟练掌握面向对象思维方式在场景

		景案例分析【A】	案例中的分析
8	2.生活中的类与对象	类与对象的概念【K】	了解类与对象的概念以及特性
9		类与对象的关系【C】	理解类与对象之间的关系
10		类的抽象方法【C】	理解场景案例中类的抽取方法
11		通过场景案例对象分析实现类的设计【A】	熟练掌握场景案例中的类的设计方法
12	3.类之间的关系	依赖关系【C】	理解类与类之间的依赖关系在案例场景中的体现
13		关联关系【C】	理解类与类之间的关联关系在案例场景中的体现
14		聚合关系【C】	理解类与类之间的聚合关系在案例场景中的体现
15		组合关系【C】	理解类与类之间的组合关系在案例场景中的体现
16		继承关系【C】	理解类与类之间的继承关系在案例场景中的体现
17		不同的类关系在场景案例中的体现【A】	能够熟练区分不同场景案例中的类与类之间的关系
18	4.UML类图	UML类图概述【K】	了解UML类图概念
19		UML类图绘制 visio【C】	掌握UML类图的组成以及 visio 工具绘制类图的方法
20		通过场景案例对象分析实现UML类图绘制【A】	熟练掌握 visio 工具绘制类图
21	5.UML时序图	UML时序图概述【K】	了解UML时序图概念
22		UML时序图绘制 visio【C】	掌握UML时序图的组成以及 visio 工具绘制时序图的方法
23		通过场景案例对象分析实现UML时序图绘制【A】	熟练掌握 visio 工具绘制时序图
24	6.C++结构程序设计	数据类型【K】	了解C++语言中的数据类型
25		变量与常量【C】	熟练掌握变量与常量的语法定义与赋值
26		基本输入输出语句【C】	熟练掌握C++语言中的基本输入输出语句
27		运算符与表达式【C】	熟练掌握不同运算符的使用场景以及使用语法
28		流程控制【A】	熟练掌握判断语句、循环语句以及中断语句的使用场景与使用语法
29		过程化代码片段的编写【A】	能够在开发工具中编写出满足简单场景需求的C++代码片段
30	7.C++函数	函数的定义与调用【C】	熟练掌握不同场景下函数的定义与调用方法
31		函数递归【C】	理解并掌握函数递归的使用场景以及调用意义

32			函数重载【C】	理解并掌握函数重载的实现条件
33			函数化代码片段的编写【A】	能够在开发工具中编写出满足简单场景需求的 C++代码片段,并能进行函数包装和调用,从而达到代码复用目的
34		8.C++数组	一维数组【C】	理解并掌握一维数组的定义、初始化以及简单的数据操作
35			字符串【C】	理解字符串的存储方式,并熟练掌握字符串的简单处理
36			集合代码片段的编写【A】	能够在开发工具中使用一维数组语法编写满足简单集合场景需求的 C++代码片段
37		9.C++容器	向量 Vector【C】	理解并掌握向量 Vector 的定义、初始化以及简单的数据操作
38			集合代码片段的编写【A】	能够在开发工具中使用向量 Vector 语法编写满足简单集合场景需求的 C++代码片段
39		10.C++指针	指针与引用的概念【K】	了解指针和引用的存储方式
40			指针运算【C】	理解地址运算以及数值运算
41			指针与引用【C】	理解并掌握指针与引用的定义与赋值方法
42			指针与数组【C】	理解指针与数组的关系,以及运用场景
43			指针与函数【C】	理解并掌握指针作为函数参数或函数返回值的设置方法
44			指针在应用场景中的代码实现【A】	能够在开发工具中使用指针与引用编写满足简单数据处理场景需求的 C++代码片段
45		11.C++ 程序调试与异常处理	断点定位【C】	了解断点的作用并能够使用 VS 工具中的断点进行问题定位
46			异常输出与处理【C】	熟练掌握 BUG 修复方法
47			代码片段的调试跟踪与处理【A】	能够通过断点调试修复错误代码
48		12.C++ 类与对象	类的定义【C】	熟练掌握类的定义语法,能够声明并实现一个简单的类
49			对象实例化【C】	熟练掌握两种对象实例化方法
50			多个对象创建的代码片段【A】	能够在开发工具中通过场景案例描述,合理抽象并设计不同的类,并创建多个不同对象
51		13.C++ 类的依赖关系	对象间依赖交互的代码片段【A】	能够在开发工具中编写满足类依赖关系场景需求的 C++代码片段
52		14.C++ 类的关联关系	对象间关联交互的代码片段【A】	能够在开发工具中编写满足类关联关系场景需求的 C++代码片段
53		15.C++ 类的组	对象间组合交互的代码片段	能够在开发工具中编写满足类组合

		合关系	【A】	关系场景需求的 C++代码片段
54		16.C++ 类的聚合关系	对象间聚合交互的代码片段【A】	能够在开发工具中编写满足类聚合关系场景需求的 C++代码片段
55		17.C++ 类的继承关系	对象间继承交互的代码片段【A】	能够在开发工具中编写满足类继承关系场景需求的 C++代码片段
56		18.C++ 成员函数	成员函数的概念【K】	了解成员函数的概念
57			成员函数的两种实现方法【C】	熟练掌握成员函数的两种实现方法
58			成员函数重载【C】	理解成员函数重载的实现条件
59			类的不同成员函数实现的代码片段【A】	能够在开发工具中，通过 C++语法实现场景案例中针对同一类的不同成员函数进行实现
60		19.C++ 构造与析构	构造函数概念【K】	了解构造函数概念以及存在意义
61			构造函数的特点【C】	理解并掌握构造函数特点，并能够熟练运用于类的初始化功能
62			析构函数概念【K】	了解析构函数概念以及存在意义
63			析构函数的特点【C】	理解并掌握析构函数特点，并能够熟练运用于类的销毁功能
64			构造与析构调用的代码片段【A】	能够在开发工具中，通过 C++语法实现场景案例中针对类的初始化与销毁的代码片段实现
65		20.C++ 封装 private 访问权限	封装的概念【K】	了解面向对象思维中的封装机制
66			private 访问权限【C】	理解并熟练掌握 private 访问权限在类中的运用
67			get 与 set 函数使用技巧【C】	理解并熟练掌握 get 与 set 函数使用技巧与应用场景
68			封装数据操作的代码片段【A】	能够在开发工具中，通过 get 与 set 函数实现场景案例中封装功能的代码片段实现
69		21.C++ 友元函数与静态成员	友元函数【C】	理解并熟练定义友元函数
70			静态成员【C】	理解静态成员概念以及应用场景
71	综合项目版本 2 (LostCraft 项目) (三子棋项目) (文字闯关项目)		类的封装实现【A】	能够在开发工具中，通过静态成员与友元函数实现场景案例中封装功能的代码片段实现
72		22.C++ 单继承	继承概念【K】	了解继承的概念以及特点
73			继承关系中三种访问权限【C】	理解并掌握继承关系中的三种访问权限的使用方法
74			不同权限数据在继承中的运用代码实现【A】	能够在开发工具中，通过不同权限数据在继承中的运用代码实现来模拟继承关系中的不同运用场景
75		23.C++ 继承中的构造与析构	继承关系中构造与析构调用规则【C】	熟练掌握继承关系中的构造与析构函数的调用规则
76			继承关系中构造与析构的代码实现【A】	能够在开发工具中，通过构造与析构函数的调用实现继承关系中的不同

				场景模拟	
77		24.C++ 静态多态	静态多态概念【K】	了解静态多态的概念及运用场景	
78			静态多态的代码实现【A】	能够在开发工具中，通过 C++语法实现场景案例中静态多态的代码片段实现	
79		25.C++ 动态多态	动态多态概念【K】	了解动态多态的概念及运用场景	
80			虚函数【C】	理解虚函数概念，熟练掌握虚函数实现语法以及运用场景	
81			纯虚函数【C】	理解纯虚函数概念，熟练掌握纯虚函数实现语法以及运用场景	
82			多态的动态链接代码实现【A】	能够在开发工具中，通过 C++语法实现场景案例中动态多态的代码片段实现	
83		26.C++ 文件与流	文件与流的概念【K】	了解文件与流的概念	
84			iostream 流【C】	熟练掌握 iostream 流的操作	
85			fstream 流【C】	熟练掌握 fstream 流的操作	
86			文件位置指针【C】	理解文件位置指针的存在意义以及实现语法	
87			文件流读写在应用场景中的代码实现【A】	能够在开发工具中，通过 C++语法实现对文件的简单读写操作	
88		27.C++ 泛型编程（模板）	模板的概念【K】	了解模板的定义与用途	
89			函数模板【C】	熟练掌握基本的函数模板的定义与调用	
90			类模板【C】	熟练掌握基本的类模板的定义与实例调用	
91			模板的代码实现【A】	能够自定义库容器实现数据对象的读写操作	
92	综合项目版本 3 （LostCraft 项目） （三子棋项目） （文字闯关项目）	28.C++Web 编程	CGI 的概念与架构图【K】	了解 CGI 的概念、架构图及 CGI 目录的配置	
93				CGI 环境变量【K】	了解 CGI 程序可访问的常见环境变量，知道每个变量名的作用
94				基于 CGI 的 HTTP 请求的代码实现【A】	能够编写简单的 CGI 程序，编译并部署到 Web 服务器中，实现一个具体网页的输出
95		29.基于 Qt 的 C++界面编程	QT 环境的配置【C】	了解 QT 的用途，能够基于 Windows 平台下载并安装 QT	
96			QT 界面库【C】	熟练掌握 QT 的常见类库模板和组件库的用法。	
97			基于 QT 的界面代码实现【A】	能够基于简单需求的 QT 用户界面开发，完成用户交互设计。	
总计	3	30	97		

六、课程教学实施框架（Implement）

《面向对象程序设计》课程采用“两段式+两项目”的为主要实施框架。以项目贯穿整个学习过程，该门课程结束时，学生需要完成2个项目，其中课程综合项目1个(两个版本)，综合评价项目1个。

课程实施采用的教学方法包括：启发式教学+案例驱动式教学+互动式教学+综合项目模拟实现+线上线下混合式教学

课程实施采用的教学环境包括线下机房以及线上自学平台。线下课程教学包括理论课程与上机实践课程，理论课程与上机实践课程均在机房进行实施。同时晚自习开放指定机房给学生，确保每个班级每周至少1次晚自习在机房练习，并安排助教进班辅导答疑。线上学习主要采用讯飞博思智慧学习平台，该平台具有浏览学习资源、观看教学视频、在线编程、在线测试等功能，同时讲师和助教对学生辅以线上QQ辅导答疑。

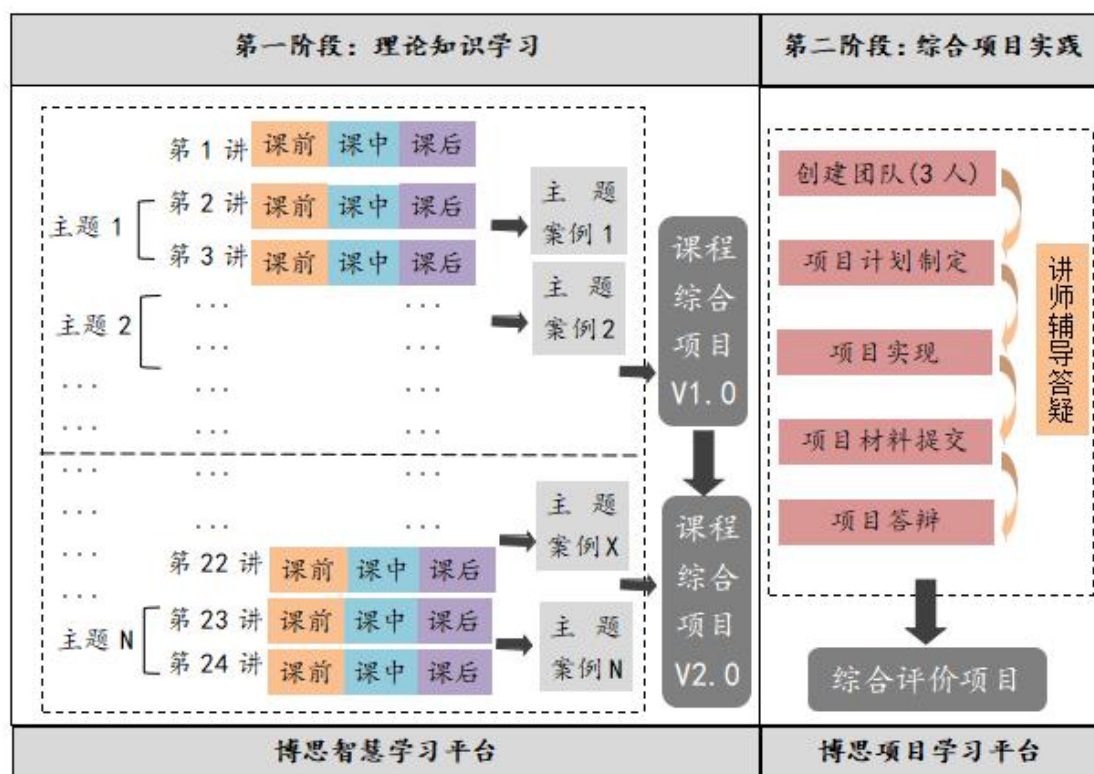


图2《面向对象程序设计》课程总体实施框架图

课程综合项目：学生通过线上学习课程综合项目，其线上资源包括项目导入（功能与知识点映射）、需求分析、功能设计（类图及交互）、开发过程（项目创建、类定义、对象交互、基本功能实现、附加功能实现与演示等）微视频教学

以及源代码等。课程综合项目包含 2 个核心功能版本，这 2 个版本将贯穿整个课程核心必修知识点，学生通过线上教学完成课程综合项目的学习。课程综合项目是第一阶段学习后的产物。

综合评价项目：学生通过构建团队形成 3 人项目小组，在第二阶段共同实现一个综合项目的功能实现，在实现过程中讲师和助教提供辅导答疑，项目开发周期为 2 周，最后以小组为单位进行项目成果材料提交与答辩，综合评价项目材料（包含源代码、答辩 PPT、项目运行视频等产物）为第二阶段学习后的产物。

七、教学评价 (Evaluation)

评价体系最终要与整个课程能力目标培养形成闭环，重新回归分析与设计。

1. 教学活动有效性评估

教学活动有效性评估数据以学生课堂教学活动参与数据为依据。例如课堂前后教师可以有针对性的发布课程任务，包括课程预习任务、课后思考练习任务等，增加师生间的教学互动。日常作业的设计区别于以往的传统形式，作业的发布、提交、批阅均通过“在线课程学习平台”进行，学生提前完成作业即可提前上交，教师通过“在线课程学习平台”进行批阅，批阅评价和评语同时也会立即反馈至学生，课程作业评价会通过可视化界面进行呈现，一方面让学生更高效的得知课程作业评价，另一方面也可以让教师迅速掌握班级学生作业完成情况。日常作业的内容也不局限于传统的简答题或简单的程序设计题，而采用阶段综合性应用项目，训练学生工程思维和编码能力，将理论知识应用到项目中去。以上出勤率、任务完成度、作业完成情况最终经由平台生成数据报表并给出评分。

类型	活动名称	开课名称: 软件1901班_面向对象程序设计_王瑞楠 课程周期: 2020/02/05-2020/07/05 (安信工19-20春季学期开课计划) 开课下学生人数: 53人 备注说明: 任务, 测试, 问卷投票, 点名及作业五项任务为百分制得分, 课堂互动种类: 提问选择器、抢答、选人、讨论、弹幕, 5项计算已发次数和参与次数, 课程学习进度及博思课程练习/自测正确率默认为开课模块下所有课程综合的学习进度及正确率
测试	《面向对象程序设计》阶段考核	
作业	软件1901班_面向对象程序设计_王瑞楠_2006031016	
任务	完成博思平台《面向对象程序设计(基础)》全部内容	
作业	软件1901班_面向对象程序设计_王瑞楠_2005301001	
作业	软件1901班_面向对象程序设计_王瑞楠_2005231443	
作业	软件1901班_面向对象程序设计_王瑞楠_2005161009	
点名	王瑞楠_20-05-12_16:22:17	
多态		
提问选择器	王瑞楠_20-05-09_16:56:19	

默认权重	25%	15%	20%	20%	10%	10%	0%	100%	扣分项				
序号	学号	姓名	点名	作业	测试	任务	课堂互动	博思课程学习进度	博思课程练习/自测正确率	平均分	作业作弊	旷课	刷题
1	91020201	张文博	100.00	82.50	73.04	93.75	100.00	100.00	36.00	91	0	0	0
2	91020201	汪宏勇	100.00	95.00	93.04	100.00	100.00	92.00	57.00	97	0	0	0
3	91020201	王小双	100.00	87.50	70.43	68.75	100.00	49.00	44.00	81	0	0	0
4	91020201	张伟莉	100.00	92.50	88.70	100.00	100.00	100.00	45.00	97	0	0	0
5	91020201	陶才坤	100.00	97.50	60.00	100.00	100.00	50.00	51.00	87	0	0	0
6	3191020	吕健康	100.00	97.50	80.87	100.00	100.00	100.00	60.00	96	0	0	0
7	91020201	张永香	100.00	97.50	68.70	100.00	100.00	100.00	34.00	94	0	0	0
8	91020201	苍涛	100.00	97.50	76.52	100.00	100.00	74.00	50.00	92	0	0	0
9	91020201	陈浩	100.00	72.50	76.52	100.00	100.00	92.00	0.00	90	0	0	0
10	91020201	陈明欢	100.00	95.00	59.13	75.00	100.00	100.00	43.00	86	0	0	0
11	91020201	杨子易	100.00	90.00	80.00	93.75	100.00	100.00	62.00	94	0	0	0
12	91020201	时豹	100.00	95.00	75.65	100.00	100.00	100.00	63.00	94	0	0	0
13	91020201	储志峰	100.00	100.00	80.00	100.00	100.00	100.00	70.00	96	0	0	0

2. 教学目标达成度评估

教学目标达成度以学生最终课程得分为依据，检验课程目标达成度，评价学生学习成果达成度。考核环节包括阶段考核、平时考核、期末考核和个人项目考核，总评成绩以百分计，满分 100 分，平时成绩由出勤、课堂提问等部分组成。各考核环节所占分值比例可根据具体情况微调，建议值及考核细则如表 5 所示。

表 5 课程考核细则表

考核环节	建议分值	考核/评价范围与标准	考核目标
阶段考核	20	(1)采取在线平台考试，题型以选择题为主，并有小型综合编程题，其中编程题为独立综合案例，学生需完成代码编写实现。	专业知识 工程能力
平时考核	25	(1)根据学生出勤和课堂参与度情况综合评分。 (2)在线学习平台资源学习情况综合评分。 (3)根据学生在线作业完成情况综合给分。	专业知识
期末考核	40	(1) 期末考试采用笔试的方式进行，按照学校教学计划放置考试周进行。期末试卷的考核范围是大纲包含的全部内容知识点。 (2) 期末考核有一票否决权，如果期末笔试的卷面成绩在 50 以下，则该门课程直接不及格，如果课程高于 50 分，则按照卷面的 40% 比例归入期末成绩中。	专业知识
个人项目考核	15	(1)个人项目考核重点在于培养和训练学生面向对象思维及建模能力，实现逻辑性较强的综合性中小型项目，考核方式为项目答辩材料评审，包括项目完成度、项目答辩等评分。	工程能力 职业素养

3. 教学资源使用率

教学资源使用率数据以学生线上学习时长、视频观看时长或次数、代码编写量、自测完成情况等多维度数据为依据，准确评估每位学生的线上学习学时是否满足基本课程设定标准，整个资源在学生自主学习过程中的使用频率以及覆盖率，并且学生可以针对课程资源进行评分并给出相关建议。



学生姓名	任务正确率 (%) ▾	练习自测正确率 (%)	进度 (%)	学习时长	自测数量
轩梦袁	93	83.2	100	49小时5分钟	167
刘玉萍	89	81.1	100	38小时17分钟	57
储志峰	88	70	100	101小时5分钟	216
方志鹏	88	84.7	100	45小时53分钟	278
尚贤	87	57.2	100	42小时4分钟	90
吕健康	87	59.5	100	15小时23分钟	134
施晨伟	84	76.6	100	37小时43分钟	77
张伟莉	84	45.2	100	47小时2分钟	92
叶子豪	83	56.4	100	22小时53分钟	78
王小双	83	44.4	100	18小时40分钟	54

面向对象思维建立

软件行业现状 进度: 100% 正确率: 42.3%	面向对象概念引入 进度: 100% 正确率: 58.7%	项目分析: 对象间行为关系 进度: 100% 正确率: 38.9%	项目分析: 类的抽象 进度: 100% 正确率: --
类与对象的概念 进度: 100% 正确率: 47.4%	类与对象的关系 进度: 100% 正确率: 73.6%	自测 进度: 94% 正确率: 77.78%	

面向对象思维建模

UML概述 进度: 100% 正确率: 87%	类图 进度: 100% 正确率: 78.9%	时序图 进度: 100% 正确率: 84.4%	用例图 进度: 100% 正确率: 84.7%
LostCraft游戏项目的类图... 进度: 100% 正确率: 76.7%	LostCraft游戏项目的时序... 进度: 100% 正确率: 68.5%	HR服务平台项目的类图设计 进度: 100% 正确率: 64.8%	HR服务平台项目的时序图... 进度: 100% 正确率: 66.8%
HR服务平台项目的用例图... 进度: 100% 正确率: 83.7%	自测 进度: 96% 正确率: 63.48%		