**《软件设计模式》课程教学大纲**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 英文课程名 | Software Design Patterns | | 总 学 时 | | 32 | 学 分 | 2 |
| 课程编码 | G126071 | | 理论教学学时 | | 24 | 适用专业 | 软件工程 |
| 课程类别（请在课程所属类别栏注明选修或必修） | 通识课程 |  | 实践  教学  学时 | 实验学时 | 8 | 先修课程 | 数据结构  Java程序设计 |
| 大类基础课程 |  | 上机学时 |  | 开课学院（部） | 计算机科学与技术学院 |
| 专业课程 | 专业选修课 | 其它 |  | 基层教学组织 | 软件工程教学团队 |

**一、课程简介**

本课程是软件工程专业的一门专业基础选修课，是关于计算机软件设计和架构分析的一门核心课程。主要讲授面向对象的软件设计中常用的设计模式（Design patterns）。这些设计模式是前人在长期的软件设计实践中的经验总结。通过学习和掌握这些设计模式可以提高软件设计者的水平，提高软件开发的可复用性、可维护性和软件的质量。该课程的先修课程是数据结构、Java程序设计或者C++程序设计等面向对象的程序设计课程，最好学习过软件工程原理课程。课程的主要内容包括设计模式的基本理念、以及涵盖创建型、结构型和行为型三个大类的23个经典的常用设计模式及其应用。该课程对培养学生的软件设计能力至关重要。

**二、教学目标**

**2.1 课程教学目标**

该课程的教学目标如下：

1. 对设计模式的相关理论形成系统的认识，了解设计模式的基本概念，掌握设计模式的描述框架，设计模式的分类方法，具备自己总结设计模式的意识。理解面向对象软件设计的基本原则，并具备在进行软件设计时应用设计原则指导设计的意识。
2. 深入掌握典型的创建型设计模式，如简单工厂模式、抽象工厂模式、工厂方法模式、单例模式等，能够在处理复杂软件工程问题时，运用创建型模式解决软件对象的创建问题，提高软件的可维护性、灵活性，满足特定的要求。
3. 深入掌握典型的结构型设计模式，如适配器模式、装饰模式等，能够在处理复杂软件工程问题时，运用结构性设计模式处理好软件中类或者对象的组合问题，从而提高软件的可维护性和灵活性。
4. 深入掌握典型的行为型设计模式，如策略模式、观察者模式等，能够在处理复杂软件工程问题时，运用行为型设计模式处理类或者对象的职责分配和交互问题。
5. 初步具备综合运用设计模式在处理复杂软件工程问题时，给出优秀的设计方案的能力。

**2.2 课程目标与毕业要求（指标点）对应关系**

【毕业要求1】**工程知识：能够将数学、自然科学、工程基础和专业知识用于解决复杂软件工程问题。**

支撑指标点1-3 能够将工程基础和专业知识用于求解软件领域复杂工程问题

【毕业要求2】**工程知识：能够将数学、自然科学、工程基础和专业知识用于解决复杂软件工程问题。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软件工程问题，以获得有效结论。**

支撑指标点2-2 能够运用数学和工程技术抽象和描述软件领域复杂工程问题

支撑指标点2-3 能够运用网络等现代技术获取信息和文献资料，并通过文献研究来分析和总结解决软件领域复杂工程问题的可能途径

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | | 毕业要求指标点 | 教学环节 | | | | |
| 课堂授课 | 实验 | 作业 | 课堂讨论 |
| **目标1：**对设计模式的相关理论形成系统的认识，了解设计模式的基本概念，掌握设计模式的描述框架，设计模式的分类方法，具备自己总结设计模式的意识。理解面向对象软件设计的基本原则，并具备在进行软件设计时应用设计原则指导设计的意识。 | **指标点2.3：**能够运用网络等现代技术获取信息和文献资料，并通过文献研究来分析和总结解决软件领域复杂工程问题的可能途径 | | √ |  | √ | √ |
| **目标2：**深入掌握典型的创建型设计模式，如简单工厂模式、抽象工厂模式、工厂方法模式、单例模式等，能够在处理复杂软件工程问题时，运用创建型模式解决软件对象的创建问题，提高软件的可维护性、灵活性，满足特定的要求。 | **指标点1.3：**能够将工程基础和专业知识用于求解软件领域复杂工程问题  **指标点2.2：**能够运用数学和工程技术抽象和描述软件领域复杂工程问题 | | √ | √ | √ | √ |
| **目标3：**深入掌握典型的结构型设计模式，如适配器模式、装饰模式等，能够在处理复杂软件工程问题时，运用结构性设计模式处理好软件中类或者对象的组合问题，从而提高软件的可维护性和灵活性。 | **指标点1.3：**能够将工程基础和专业知识用于求解软件领域复杂工程问题  **指标点2.2：**能够运用数学和工程技术抽象和描述软件领域复杂工程问题 | | √ | √ | √ | √ |
| **目标4：**深入掌握典型的行为型设计模式，如策略模式、观察者模式等，能够在处理复杂软件工程问题时，运用行为型设计模式处理类或者对象的职责分配和交互问题。 | **指标点1.3：**能够将工程基础和专业知识用于求解软件领域复杂工程问题  **指标点2.2：**能够运用数学和工程技术抽象和描述软件领域复杂工程问题 | | √ | √ | √ | √ |
| **目标5：**初步具备综合运用设计模式在处理复杂软件工程问题时，给出优秀的设计方案的能力。 | **指标点2.3：**能够运用网络等现代技术获取信息和文献资料，并通过文献研究来分析和总结解决软件领域复杂工程问题的可能途径 | |  | √ | √ |  |

**三、课程教学内容及学时分配**

**1．理论教学安排**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 章节或知识点(模块) | 教学内容 | 学时分配 | 教学要求  (应明确教学重点、难点和教学方法) | 学生任务 | |
| 作业要求 | 其他要求(自学/讨论） |
| 1 | 设计模式简介 | 1. 何为软件设计 2. 设计体验 3. 什么是设计模式 4. 学习设计模式的作用 | 2 | 重点：让学生体会到设计的差异对软件开发的影响，通过设计体验使学生体会到软件开发不是简单的编程，运用科学的方法对系统进行抽象描述和设计对软件开发的质量具有很大的影响，从而认识到学习设计模式的重要性。  难点：使学生的思维提升到一个新的层次。  教学方法：讲授+案例+代码演示 | 上网查阅资料，获取设计模式的网上资源 | 对学习设计模式的作用开展讨论 |
| 2 | 面向对象规范与UML基础 | 1. 需求变化 2. 传统的应对需求变化的方法及其不足 3. 面向对象程序设计的规范 4. UML基础 | 2 | 重点：要求学生对面向对象的设计思想有较深入的认识，能够规范的使用UML建模，从而具备对软件设计问题进行描述的基本能力。  难点：面向对象设计思想的理解。  教学方法：讲授+案例 | 复习Java面向对象的程序设计的主要内容，UML类图绘制 | 类的关联关系细分 |
| 3 | 面向对象的设计原则 | 1. 开闭原则 2. 面向抽象原则 3. 优先使用组合实现复用的原则 4. 高内聚低耦合原则 5. 原则间的关系及应用 | 2 | 重点：要求学生理解原则的含义，以及在程序设计时如何运用和体现这些设计原则；了解原则之间的关系，形成在设计中运用原则的意识；理解这些设计原则为进一步学习和理解设计模式奠定了基础。  难点：开闭原则。  教学方法：讲授+案例+代码演示 | 思考各原则的关系，分析设计体验项目解决方案中原则的应用情况 | 为什么开闭原则是目标，其他原则是手段？ |
| 4 | Strategy模式 | 1. 策略模式及其应用 | 2 | 重点：要求学生掌握策略模式的应用场景，解决方案和关键特征；使学生掌握Strategy模式在复杂软件设计中的用法，使用策略模式处理好复杂软件工程问题中需要动态灵活地替换系列算法的问题。  难点：Jar包中的策略模式。  教学方法：讲授+案例+代码演示+课堂练习 | 查找并给出应用策略模式的案例 | 什么时候需要运用策略模式？ |
| 5 | Observer模式 | 1. 观察者模式及其应用 | 2 | 重点要求学生掌握观察者模式的应用场景，解决方案和关键特征，使学生掌握Observer模式在复杂软件设计中的用法，使用观察者模式处理好复杂软件工程问题中多个类或对象依赖某个类或对象的状态变化的问题。  难点：观察者模式的使用场景和范围。  教学方法：讲授+案例+代码演示+课堂练习 | 查找并给出应用观察者模式的案例 | 什么时候需要运用观察者模式？ |
| 6 | 工厂模式 | 1. 简单工厂模式 2. 抽象工厂模式 3. 工厂方法模式 4. 各工厂模式的关键特征、区别及选用 | 2 | 重点：要求学生掌握工厂模式的应用场景，解决方案和关键特征；使学生掌握工厂模式在复杂软件设计中的用法，处理复杂软件工程问题时，能够运用创建型模式解决软件对象的创建问题，提高软件的可维护性、灵活性，满足特定的要求。  难点：不同对象创建方式的选择。  教学方法：讲授+案例+代码演示+课堂练习 | 查找并给出应用工厂模式的案例 | 什么时候需要运用工厂模式？ |
| 7 | 单例模式与适配器模式 | 1. 单例模式及其应用 2. 适配器模式及其应用 | 2 | 重点：要求学生掌握单例模式及适配器模式的应用场景，解决方案和关键特征，使学生掌握单例模式和适配器模式在复杂软件设计中的用法，增强在软件设计过程中软件复用的能力。  难点：单例和适配器模式的使用场景和范围。  教学方法：讲授+案例+代码演示+课堂练习 | 查找并给出应用单例模式的案例 | 单例模式与适配器模式的应用场景。在多线程下如何确保单例模式的实现 |
| 8 | Decorator模式 | 1. 装饰模式及其应用 | 2 | 重点：要求学生掌握装饰模式的应用场景，解决方案和关键特征。在复杂软件设计中，能够洞察装饰模式的应用条件，使用装饰模式，通过类或对象的组合实现对算法的动态扩展，从而提高软件的可维护性和灵活性。  难点：装饰模式的使用场景和范围。  教学方法：讲授+案例+代码演示+课堂练习 | 查找并给出应用装饰模式的案例 | 装饰模式为什么可以有效解决类爆炸的问题？ |
| 9 | 设计模式研讨 | 学生做小组演讲并开展课堂讨论 | 6 | 重点：培养学生的自主学习能力，表达能力和交流沟通等方面能力。使学生掌握自学的模式在复杂软件设计中的用法，提高面向对象的软件设计能力。  难点：如何让学生更有效地进行资料查找和表达。  教学方法：课堂提问和点评 | 检索资料，自学模式，制作ppt | 针对同学的演讲提问，并讨论 |
| 10 | 课程总结 | 1. 设计模式的分类及相互关联 2. 面向对象的设计原则总结 3. 设计模式的选用 | 2 | 重点：对设计模式有一个全面的了解，能够较合理地选用设计模式解决实际设计问题，对面向对象的设计原则形成较为深刻的认识。使学生综合掌握各种模式在复杂软件设计中的用法，对复杂软件工程问题给出优良的设计方案。  难点：模式之间的关联。  教学方法：讲授 | 研读23个模式的关联图，分析其特点、异同、关联性及选用条件。 |  |

2．实践教学安排

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 项目名称 | 学时 | 类型 | 每组人数 | 教学要求  (应明确教学重点、难点和教学方法) | 学生任务 | |
| 作业要求 | 其他要求(自学/讨论） |
| 1 | Strategy模式的应用 | 2 | 设计型 | 1 | 重点：要求学生运用策略模式解决给定的设计问题，给出设计方案并加以实现。  难点：如何更好地实现开闭原则。  教学方法：上机+现场教学和讨论 | 在具体项目中运用策略模式设计解决方案 | 无 |
| 2 | Observer模式的应用 | 2 | 设计型 | 1 | 重点：要求学生运用观察者模式解决证券交易软件中，一个股票的各种分析指标随股票价格波动而同时变化的设计问题，给出设计方案并加以实现。  难点：如何用设计模式解决复杂工程问题。  教学方法：上机+现场教学和讨论 | 在具体项目中运用观察者模式设计解决方案 | 无 |
| 3 | 扩展单例模式应用 | 2 | 设计型 | 1 | 重点：要求学生在具体项目中运用扩展单例模式完成类似数据库连接池的设计方案，并加以实现。  难点：如何用设计模式解决复杂工程问题。  教学方法：上机+现场教学和讨论 | 在具体项目中运用扩展单例模式完成类似数据库连接池的设计方案 | 无 |
| 4 | Decorator模式的应用 | 2 | 设计型 | 1 | 重点：要求学生在具体项目中运用装饰模式完成聊天软件中屏蔽敏感词汇功能的设计方案，并加以实现。  难点：如何用设计模式解决复杂工程问题。  教学方法：上机+现场教学和讨论 | 在具体项目中应用装饰模式设计解决方案 | 无 |

**四、考核方式及成绩评定方式**

该课程的考核强调过程化考核。其总成绩分为进程性成绩和期末考试成绩两大部分，而进程性成绩主要考核学生的实验能力、自主学习能力和对课程的总结能力等几个方面。各个环节所占比例及基本要求如下：

（1）期末考试：占总成绩的50%。要求：试卷难度适中，填空、选择、判断、名词解释、简答、论述题等基础性题目严格控制比例，加大分析设计题目的比例，重在考查学生运用知识解决复杂工程问题的能力。

（2）实验：占平时成绩的约35%。要求：课程设置四次课内实验，每次实验提前布置给学生，要求学生通过课外查阅文献资料、熟悉相关软件设计的背景，领会设计意图，熟悉要运用的设计模式，实现语言，进行方案的预设计和实现，以保证实验效果。通过实验，加强学生对复杂软件工程问题的分析能力、资料搜索能力、和运用设计模式解决软件设计问题的能力。

（3）自主学习与表达：占平时成绩的约10%。要求：教师将学生分组，每组布置一个设计模式让学生自学，要求他们自己查找资料和应用案例，在自学讨论的基础上制作PPT，编写示例代码并进行小组展示演讲、课堂讨论和点评。重点考察学生的自主学习能力、团队协作能力、语言表达与沟通能力。小组成员的成绩按小组自己提供的各成员贡献度分摊，尽量避免吃大锅饭的情况发生。

（4）小论文：占平时成绩的约5%。要求：教师布置学生完成一篇有关设计模式的小论文，可以是对学习的设计模式进行总结也可以就某个设计模式的应用进行深入的探讨，重点考察学生对课程知识的总结能力、学习体会以及书面表达能力。

期末考试可采用开卷或者“一页开卷”的形式，以便将学生从死记硬背中解脱出来，将更多的精力放在知识的理解和灵活运用上，同时也能更真实地考核学生的实际水平。

**五、教材、课程网址及参考书目**

教 材： 《Java设计模式》，刘伟，清华大学出版社，2018年7月

参考书：

【1】《设计模式之禅（第2版）》. 秦小波. 机械工业出版社出版, 2014年2月。

【2】《设计模式解析（第2版）》. Alan Shalloway，James R.Trott，人民邮电出版社，2010年12月。

【3】《设计模式初学者指南》．Alan Holub．机械工业出版社，2006。

【4】《大话设计模式》.吴强. 企业管理出版社,2010年7月。

**执笔者：王婷、黄洪**

**审核者：张繁**

**课程教学团队成员：王婷、黄洪、黄玉娇**