

第一、二章
第四章 项目启动
第五章
第六章
第七章
第八章
第九章
第十一章
第十二章
第十三章
第十四章
第十五章
第十六章
第十七&十八章
第十九章
第二十&二十章
第二十二&二十三章
附录

## 第一、二章

---

### 第一、二章

---

- 名词解释
  - 软件工程
- 简答：
  - 从1950s~2000s之间的特点

- 软件工程：
  - 用系统的、规范的、可量化的方法开发、测试和维护软件

- 对上述活动的研究
- 特点：
  - 50：科学计算，以机器为中心进行编程，像生产硬件一样生产软件
  - 60：业务应用（批量数据处理和事务计算）；软件不同于硬件，用软件工艺方法生产软件
  - 70：结构化方法，瀑布模型，强调规则和纪律
  - 80：追求生产力最大化；现代结构化方法/面向对象编程广泛应用，重视过程的作用
  - 90：企业为中心的大规模软件系统开发，追求快速开发、可变更性和用户价值；web应用出现
  - 00：大规模web应用，大量面向大众的web产品；最求和快速开发、可变更性和用户价值

## 第四章 项目启动

### 第四章 项目启动

- 如何管理团队？
  - 在实验中采取了哪些办法？有哪些经验？
  - 团队结构有哪几种？
- 质量保障有哪些措施？
  - 结合实验进行说明
- 配置管理有哪些活动？
  - 实验中是如何进行配置管理的？
- 示例：（2010）结合实验，说明一个项目的质量保障包括哪些活动？

- 管理团队
  - 建立团队章程
  - 和谐沟通
  - 持续成功
  - 避免团队杀手
- 团队结构
  - 主程序员团队
  - 民主团队
  - 开发团队
- 质量保障措施
  - 需求开发：需求评审、需求度量
  - 体系结构设计：体系结构评审、集成测试（持续集成）

- 详细设计：详细设计评审、设计度量集成测试（持续集成）
- 软件构造：代码评审、代码度量、测试（测试驱动、持续集成）
- 测试：软件测试、测试度量
- 配置管理的活动：
  - 标识配置项
  - 版本管理
  - 变更控制
  - 配置审计
  - 状态报告
  - 软件发布管理

## 第五章

---

### 第五章

---

- 名词解释
  - 需求
- 区分需求的三个层次
  - 给出一个实例，给出其三个层次例子
  - 对给定的需求示例，判定其层次
    - 例如课程实验/ATM/图书管理...
- 掌握需求的类型
  - 对给定的实例，给出其不同类型的需求例子
  - 对给定的需求示例，判定其类型
    - 例如课程实验/ATM/图书管理...

## 第六章

---

## 第六章

---

- 为给定的描述
  - 建立 用例图
  - 建立 分析类图
  - 建立 系统顺序图
  - 建立 状态图
- 用例图
  - 明确目标
  - 寻找参与者
  - 寻找用例
  - 细化用例
- 分析类图
  - 识别候选类
  - 筛选候选类、确定概念类
  - 识别关联
  - 确定重要属性

## 第七章

---

## 第七章

- 为什么需要需求规格说明？结合实验进行说明
  - 对给定的需求示例，判定并修正其错误
    - 对给定的需求规格说明片段，找出并修正其错误
  - 对给定的需求示例，设计功能测试用例
    - 结合测试方法
- 
- 需求规格说明：
    - 简介
    - 精确 ✨
    - 易读
    - 易修改
- 
- 为下列描述建立用例模型（6分），要求明确给出建模过程（4分）。
  - 现在需要开发一个简化了的大学图书馆系统，它有几种类型的借书人，包括教职工借书人、研究生借书人和本科生借书人等。借书人的基本信息包括姓名、地址和电话号码等。对于教职工借书人，还要包括诸如办公室地址和电话等信息。对于研究生借书人，还要包括研究项目和导师信息等。对于本科生借书人，还要包括项目和所有学分信息等。
  - 图书馆系统要跟踪借出书本信息。当一个借书人捧着一堆书去借书台办理借书手续时，借出这个事件就发生了。随着时间的过去，一个借书人可以多次从图书馆中借书。一次可以借出多本图书。
  - 如果借书人想要的书已被借出，他可以预约。每个预约只针对一个借书人和一个标题。预约日期、优先权和完成日期等信息需要维护。当借书完成，系统会将这本书与借出联系起来。
  - 借书人根据图书馆的信息来检索书名，同时检索这本书是否可以被借出。如果一本书的所有副本都被借出了，那么借书人可以根据书名预订这本书。当借书人把书拿到借书台的时候，管理员可以为这些书办理归还手续。管理员要跟踪新书到达的情况。
  - 图书馆的管理者有属于自己的活动。他们要分类打出关于书的标题的表格，还要在线检查所有过期未还的书，也标出来。而且，图书馆系统还可以从另外一个大学的数据库中访问和下载借书人的信息。

- 下面是一段需求描述，请依据其建立ATM机系统的领域类图（即分析类图）。
- A银行计划在B大学开设银行分部，计划使用ATM机提供全部服务。ATM系统将通过显示屏幕、输入键盘（有数字键和特殊符号键）、银行卡读卡器、存款插槽、收据打印机等设备与客户交互。客户可以使用ATM机进行存款、取款、余额查询等操作，它们对帐户的更新将交由账户系统的一个接口来处理。安全系统将为每个客户分配一个PIN码和安全级别。每次事务执行之前都需要验证该PIN码。在将来，银行还计划使用ATM机支持一些常规的操作，例如地址和电话号码修改。

## 第八章

---

### 第八章

---

- 名词解释
    - 软件设计
  - 软件设计的核心思想是什么？
  - 软件工程设计有哪三个层次？各层的主要思想是什么？
- 
- 软件设计：软件设计是关于软件对象的设计，是一种设计活动，具有设计的普遍性。软件设计既指软件对象实现的规格说明，包含软件设计和软件原型，也指产生这个规格说明的过程。即软件设计工程师以对象的目标和用户的要求为指引，在一定的客观约束条件下，通过分析、研究、建模等活动进行软件设计。
  - 软件设计的核心思想：

- 抽象：纵向上聚焦各子系统之间的接口
- 分解：横向上将系统分割为几个相对简单的子系统以及各子系统之间的关系
- 三个层次：
  - 高层：基于反映软件高层抽象的结构层次，描述系统的高层结构、关注点和设计决策
  - 中层：更加关注组成构建的模块划分、导入导出、过程之间的调用关系或类之间的协作
  - 底层：深入模块和类的内部，关注具体的数据结构、算法、类型、语句和控制结构等。

## 第九章

---

### 第九、十章

---

- 体系结构的概念
- 体系结构的风格的优缺点
- 体系结构设计的过程？
- 包的原则
- 体系结构构建之间接口的定义 (\*)
- 体系结构开发集成测试用例
  - Stub和Driver

## 第十一章

---

## 第十一章

- 名词解释
  - 可用性
- 能够列出至少5个界面设计的注意事项，并加以解释
  - 例子\*\*违反了哪些条界面设计原则
- 精神模型、差异性
- 导航、反馈、协作式设计



分析上图：它们分别体现（或违反）了哪些人机交互设计原则？请详细解释这些原则（10分）。





该界面很好的运用了哪些人机交互原则？请至少列出3条并进行详细的解释

## 第十二章

### 第十二章

- 详细设计的出发点
- 职责分配
- 协作
- 控制风格
- 给定分析类图、系统顺序图和设计因素描述
  - 建立设计类图
  - 或者详细顺序图
- 协作的测试
  - MockObject

## 第十三章

# 第十三章

<ul style="list-style-type: none"><li>名词解释：解释耦合与内聚</li><li>耦合与内聚<ul style="list-style-type: none"><li>对例子**，说明它们之间的耦合程度与内聚，给出理由</li></ul></li><li>信息隐藏<ul style="list-style-type: none"><li>基本思想</li><li>两种常见的信息隐藏决策</li><li>对例子**，说明其信息隐藏程度好坏</li></ul></li></ul>	<div>1. Coincidental cohesion 2. Logical cohesion 3. Temporal cohesion 4. Procedural cohesion 5. Communicational cohesion 6. Functional cohesion 7. Informational cohesion</div> <div>Low cohesion - bad</div> <div>High cohesion - good</div> <div>5. Content Coupling (High Coupling - Bad) 4. Common Coupling 3. Control Coupling 2. Stamp Coupling 1. Data Coupling (Low Coupling - Good) (Remember: no coupling is best!)</div>	第十二章
--	--	------

# 第十四章

<div>第十四章</div> <ul style="list-style-type: none"><li>Principles from Modularization<ul style="list-style-type: none"><li>1: 《Global Variables Consider Harmful》</li><li>2: 《To be Explicit》</li><li>3: 《Do not Repeat》</li><li>4: Programming to Interface (Design by Contract)</li><li>5:The Law of Demeter</li><li>6: Interface Segregation Principle(ISP)</li><li>7: Liskov Substitution Principle (LSP)</li><li>8: Favor Composition Over Inheritance</li><li>9: Single Responsibility Principle</li></ul></li><li>对给定的示例，发现其所违反的原则，并进行修正</li></ul>	第十二章第十
--	--------

三章

# 第十五章

## 第十五章

- 信息隐藏的含义
- 封装
- OCP
- DIP



分析下图请说明：（1）它正确运用了哪条面向对象设计原则，请详细解释这条原则（5分）；（2）User类的内部是哪种层次的内聚？请详细描述这个层次内聚的特征（5分）；（3）它违反了哪条面向对象设计原则，请详细解释这条原则（5分）。

## 第十六章

## 第十六章

---

- 如何实现可修改性、可扩展性、灵活性
- 策略模式
- 抽象工厂模式
- 单件模式
- 迭代器模式
- 给定场景，应用设计模式并写出代码
- 给出代码，要求用设计模式改写

## 第十七&十八章

---

### 第十七、十八章

---

- 构造包含的活动
- 名词解释
  - 重构、测试驱动开发、结对编程
- 给定代码段示例，对其进行改进或者发现其中的问题
  - 简洁性/可维护性
  - 使用数据结构消减复杂判定
  - 控制结构
  - 变量使用
  - 语句处理
  - How to write unmaintainable code
  - 防御与错误处理
- 单元测试用例的设计
- 契约式设计
- 防御式编程
- 表驱动

## 第十九章

---

### 第十九章

---

- 掌握白盒测试和黑盒测试的常见方法，并进行优缺点比较
- 能解释并区别白盒测试三种不同的方法：语句覆盖、分支覆盖和路径覆盖
- 给出一个场景，判断应该使用哪种测试方法，如何去写（\*）
  - 对给定的场景和要求的测试方法，设计测试用例
    - 给出功能需求，则要求写功能测试用例
    - 给出设计图，则要求写集成测试用例，Stub and Driver
    - 给出方法的描述，则要求写单元测试用例，Mock Object
    - JUnit基本使用方法

## 第二十&二十章

---

### 第二十、二十一章

---

- 如何理解软件维护的重要性？
- 开发可维护软件的方法
- 演化式生命周期模型
- 用户文档、系统文档
- 逆向工程、再工程

## 第二十二&二十三章

---

## 第二十二、二十三章

---

- 软件生命周期模型
- Build-and-fix model
- Waterfall model
- Iterative Models
  - Incremental model
    - Incremental Delivery
  - Evolutionary
    - Evolutionary development
    - Prototyping
    - Spiral Model
- 解释与比较不同过程模型（要求、特征描述、优点、缺点）
- 对给定的场景，判定适用的开发过程模型
- 软件工程知识体系的知识域

## 附录

---

### 题型和数量

---

- 数量
  - 8-10道大题
- 考试题型
  - 中文试题
  - 全部是大题