**英文版《软件工程》教学内容回顾2018下**

**(下述问题仅是课件中的主要部分回顾，问题答案以课件为主要参考)**

**Chapter01**

1. **SE的定义、目的、方法及作用:**

**SE：**在将有关软件开发与应用的概念科学体系化的基础上，研究如何有计划、有效率、 经济的开发和利用能在就算机上正确运行的软件的理论和技术的工程方法学，一些开发和维护 软件的方法、过程、原则。**是一个系统工程，既有对技术问题的分析与综合，也有对开发过程和参与者的管理。**

**SE的方向：**面向对象模式，结构化模式，基于过程的模式等

**SE的作用：**付出较低的开发成本，达到要求的软件功能，取得较好的软件性能，开发的软件易于移植，需要较低的维护费用，能按时完成开发工作，及时交付使用。

**//** 开发模式（paradiam）

1. **说明错误、缺陷、失败的含义与联系。（请举例说明）**

**错误：** 软件生产中认为的错误（误解需求，错误代码等）

**缺陷：** 功能实现中的问题 （一个错误----几个故障）（静态存在）

**失败：** 运行软件失败（因缺陷）（动态存在）

1. **软件质量应从哪几个方面来衡量？论述之。\***

**产品的质量**

**过程的质量**

**商业环境背景下的质量**

// 软件系统的系统组成。

1. **现代软件工程大致包含的几个阶段及各个阶段文档。**

（1）需求分析：:《SRS》软件需求规格说明书

  （2）系统设计：系统结构图SAD

  （3）程序设计：包括模块功能算法与数据描述

  （4）程序实现：主要包括编程的代码和注释

  （5）单元测试：模块测试与性能测试 测试报告

  （6）集成测试：按照结构图SAD进行测试产生测试报告

  （7）系统测试：按SRS对系统总体功能进行测试

  （8）系统提交：交付产品

  （9）维护：修改软件的过程，为满足改错或新需求

//使现代SE实践发生变化的（七个）关键因素是什么？

1. **什么是软件过程？软件过程的重要性是什么？包含几个阶段？**

**软件过程:** 软件开发活动中的各种组织及规范方法

1. **什么是重用、抽象等现代软件工程主要概念？\*\***

**重用:** 重复采用以前开发的软件系统中具有共性的部件, 用到新的开发项目中去.(注: 这里的重用绝不仅仅是源代码的重用)

**抽象:**

**Chaoter02**

1. **什么是软件过程？软件过程的重要性是什么？软件生命周期？\*\***

软件过程： 软件开发活动中产生某种期望结果的一系列有序任务，涉及活动、约束和资源.

通用性 一致性 结构性 自我指导性

 重要性：（1）它强制活动具有一致性和一定的结构，使程序的集合组合起来以产生满足目标和标准的产品，（2）过程结构允许我们分析、理解、控制和改进组成过程的活动，并以此来指导我们的行动（3）它能使我们获取经验并把它创收给他人。

1. **瀑布模型及各阶段文档，优缺点？**

《SRS》软件需求规格说明书

系统设计文档《SAD》

模块功能算法和数据描述文档

源程序和注释

单元测试报告

系统测试报告

验收测试报告

维护报告

优点：

描述规范软件开发活动，每一个过程活动都有与其相关联的里程碑和可交付产品，; （项目经理可以衡量进展情况）

简单性：很容易向用户解释

其他复杂模型的基础： 如加入反馈循环及额外的活动

缺点：

面临软件变动时, 该模型无法处理实际过程中的重复开发问题（软件是一个

创造的过程, 不是一个制造的过程）

当时的文档转换有困难

1. **原型的概念与用途。**
2. **论述分阶段开发模型的含义, 其基本分类及特点是什么？**
3. **螺旋模型四个象限的任务及四重循环的含义？**

// ------ 习题2， 3。

// 在所有的软件开发过程模型中，你认为哪些过程给予你最大的灵活性以应对需求的变更？

1. **什么是UP， RUP，进化式迭代等市场流行的过程模型?**

**Chapter03**

什么是项目进度？活动？里程碑？

如何计算软件项目活动图的关键路径？（习题2，3）冗余时间？最早和最迟开始时间（课堂习题讲解）

// 软件团队人员应该具备的能力是什么？

软件项目团队组织的基本结构？

专家估算法的大致含义？算式估算法的大致含义？

试述COCOMO模型的三个阶段基本工作原理或含义。

什么是软件风险？ 了解主要风险管理活动？有几种降低风险的策略？

找出课后练习题图3.23和3.24的关键路径。

**Chapter04**

1. **需求的含义是什么?**

是对来自用户的关于软件系统的期望行为的综合描述, 涉及系统的对象、状态、约束，功能等

1. **需求作为一个工程，其确定需求的过程是什么？**

原始需求获取

问题分析

规格说明草稿

需求核准

软件需求规格化说明 SRS

1. **举例说明获取需求时，若有冲突发生时，如何考虑根据优先级进行需求分类。**

绝对要满足的需求（必须的）

非常值得要的但并非必须的需求（值得要的）

可要可不要的需求（可选的）

// 如何使需求变得可测试？（sidebar4.4）

1. **需求文档分为哪两类？**

****需求定义****，它面向的是业务相关的人员，例如：委托人，客户以及用户；

****需求规格说明****，它面向的是技术性人员，例如：设计人员、测试人员以及项目经理。

1. **什么是功能性需求和非功能性需求/质量需求？ 设计约束？过程约束？如何区分？**

**功能需求：**描述系统内部功能或系统与外部环境的交互作用,涉及系统输入应对,实体状态变化,输出结果,设计约束与过程约束等.

**设计约束**：

物理环境：对环境或设备的限制等（安装及环境要求等）

接口：涉及输入输出的限制或约束条件. (输入格式预定等)

用户：使用者的基本情况（限定几种类型的用户）

**过程约束**：

资源 ：材料、人员技能或其它。

文档 ：类型、数量或其它。（涉及其针对性及要求等）

标准 ：比如阅读文档时的用户指派标准。

其他 ：什么原因会导致从工资单列表中删除某雇员？

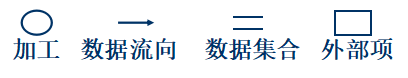
**非功能性需求/质量需求：**

描述软件方案必须具备的某些质量特征，如系统性能，安全性，反应时间。

// 需求的特性？（正确性、一致性、完整性）。

1. **了解DFD图的构成及画法。**

**数据流图DFD**：



// 在需求原型化方面，什么是抛弃型原型？什么是演化型原型？

// 用DFD图简单描述ATM机的工作原理（主要功能和数据流）（习题7）

**Chapter05**

1. **什么是软件体系结构？设计模式？设计公约？设计？ //概念设计？技术设计？**

**体系结构 :**一种软件解决方案，用于解释如何将系统分解为单元，以及单元如何相互关联，还包括这些单元的所有外部特性。

**设计模式:** 一种针对单个软件模块或少量模块而给出的一般性解决方案，它提供较低层次的设计决策。(此设计决策低于体系结构) (注: 此处为说明,不是定义)

**设计公约:**一系列设计决策和建议的集合，用于提高系统某方面的设计质量

**设计:** 将需求中的问题描述转变成软件解决方案的创造性过程

**概念设计**：告诉客户什么系统会做（软件架构和功能）

**技术设计**：告诉程序员如何系统会做----软件功能和界面 实现方法（编码器的参考 文件）

1. **软件设计过程模型的几个阶段？**

**初始建模:**尝试可能的分解：根据需求描述的系统的关键特性等确定软件体系结构风格，进行系统级别的决策。  
**分析** :主要关注软件系统的质量属性

**文档化:** 确定各个不同的模型视图

**复审 :**检查文档是否满足所有功能及质量需求。

**正式的 <SAD>** ：软件体系结构文档

// 三种设计层次极其关系？

//什么是模块化？什么是抽象？

1. **论述设计用户界面应考虑的问题。**

**寓意/比喻** :基本术语、图像和概念等

**思维模型:** 数据、功能、任务的组织与表示

**模型的导航规则:** 如何在数据、功能、活动和角色中移动及切换

**外观:**

**感觉**

**文化差异问题:** （信仰，价值观，道德规范，传统，风俗，传说）

国际设计/无偏差设计 剪裁界面

**用户爱好问题:** 制作替代界面，供有不同偏好的人选择

1. **5.5节----模块独立性----耦合与内聚的概念及各个层次划分？\***

**耦合:** 两个软件部件之间的相互关联程度

**内聚:** 软件部件内部各组成成分的关联程度

1. **举例说明耦合与内聚的基本分类。以及各个分类的含义与特征\***

（非直接耦合） ：模块相互之间没有信息传递

（数据耦合） ：模块间传递的是数据

（特征耦合）：模块间传递的是数据结构

（控制耦合）：模块间传递的是控制量

（公共耦合）：不同模块访问公共数据

（内容耦合）：一个模块直接修改另一个模块(A模块直接调用B模块的私有 数据, 或直接转移到B模块中去)

（偶然性内聚）：不相关的功能, 过程,数据等出现在同一个部件中.

（逻辑性内聚）：逻辑上相关或相似的功能或数据放置在同一个部件内

（时间性内聚）：部件各部分要求在同一时间完成

（过程性内聚）：各部分有特定次序。

（通讯性内聚）：各个部分访问共享数据

（顺序性内聚）：各部分有输入输出关系。

（功能性内聚）：各部分组成单一功能

1. **软件过程中复审的概念，设计复审的重要性。\***

**复审:** 检查文档是否满足所有功能及质量需求。

**Chapter06**

// 什么是面向对象？OO有几个基本特征？如何使用高级语言实现这些基本// 特征？了解并使用高级语言的OO基本编程方法和技巧。

什么是设计模式？

OO设计的基本原则？

OO开发有何优势？

OO开发过程有几个步骤？

**了解用例图**的组成和画法，用例的几个要素的含义。 //掌握用例图的实例解析方法

用例图、类图等对面向对象的项目开发的意义是什么？

**熟悉类图**中各个类之间的基本关系分类及其含义，状态图的含义及用途。

熟悉用例图、类图、状态图等的组成和画法。

**了解**UML其他图示结构的基本用途。

**Chapter07**

//为什么说编码工作是纷繁复杂甚至令人气馁？

一般性的编程原则应该从哪三个方面考虑？

//论述编码阶段实现某种算法时所涉及的问题。

在编写程序内部文档时，除了HCB外，还应添加什么注释信息？注意什么？

什么是极限编程(XP)? 以及派对编程？

**Chapter08**

// 产生软件缺陷的原因？

// 将软件缺陷进行分类的理由？

有几种主要的缺陷类型？

什么是正交缺陷分类？

测试的各个阶段及其任务？（图8.3）

// 测试的态度问题？（为什么要独立设置测试团队？）

**掌握**测试的方法----黑盒、白盒的概念？

什么是单元测试？ 什么是走查和检查？

黑盒白盒方法各自的分类？测试用例的设计和给出方法（结合补充材料）

黑盒白盒方法的分类，各种覆盖方法等。（课件和补充课件）

如何面对一个命题，设计和给出测试用例的问题。（课件）

------课堂练习的测试题目和讲解内容

集成测试及其主要方法的分类？（驱动，桩的概念）

// 传统测试和OO测试有何不同？OO测试有何困难？

// 测试计划涉及的几个步骤？ （**了解**）

**Chapter09**

系统测试的主要步骤及各自含义？（图9.2）

// 什么是系统配置？软件配置管理？ // 基线？（或见课件）

什么是回归测试？

功能测试的含义极其作用？

功能测试的基本指导原则？

性能测试的含义与作用？

性能测试的主要分类？

// 什么是可靠性、可用性和可维护性？

确认测试概念，确认测试分类？（基准测试和引导测试）

什么是alpha测试？β测试？

什么是安装测试？

**注意：**

**每一章节的开头中，大的概念性问题是如何引入的？，其讨论请见课件。**

**试卷答题须知：**

**1．软件工程课程出题覆盖范围比较广泛，考察手段有灵活和多样化特点。**

**2．有考察学生软件工程运用能力的题目存在，难题占的分数很少。其他很多是送分的概念型题目。**

**3．鉴于有难度的题目占极少部分，可以根据时间安排，先回答别的题目。**

**4．所有题目都要写到试卷的指定位置，以免流水阅卷时有遗漏。**