

软件工程复习

1. 第一章 软件工程概述

- 软件工程学三大要素：方法，工具，过程
- 软件危机：指在计算机软件的开发和维护中所遇到的一系列严重问题
- 软件工程：是指导计算机软件开发和维护的一门工程学科
- 软件生命周期：软件定义，软件开发，软件维护09
- 软件生存周期就是从提出软件产品开始，直到该软件产品被淘汰的全过程
- 软件定义的三个时期：问题定义，可行性研究，需求分析
- 结构化方法：按用户至上的原则，结构化，模块化，自顶向下地对系统进行分析与设计
- 软件生存期模型：
 - 瀑布模型：
 - 六个基本活动：需求分析验证，规格说明验证，设计验证，编码验证，综合测试，维护
 - 至上而下，相互衔接地固定次序
 - 增量模型：
 - 迭代的，强调每一个在增量均发布一个可操作地模型
 - 快速原型模型：
 - 增量模型地另一种形式，在该原型地基础上，逐渐完成整个系统地开发工作
 - 喷泉模型：
 - 以用户需求为动力，以对象为驱动地模型，主要是面对对象地开发
 - 螺旋模型：
 - 瀑布模型和快速原型模型结合，强调风险分析，适合大型项目
 - 缺点：风险驱动，要求软件开发人员具有更丰富地经验
 - 统一过程原型：
 - 以用例驱动为特点，以体系结构为核心，以迭代及增量为策略地软件过程模型。应用于面对对象项目
 - 敏捷开发模型：
 - 以用户需求进化为核心，采用迭代，循序渐进地方式

2. 第二章 结构化分析

- 结构化分析的三个步骤：问题定义，可行性研究，需求分析
- 可行性研究的内容：技术可行性，经济可行性，法律进行性
- 需求分析阶段：获取需求，分析需求，定义需求，验证需求
- 结构化分析的框架：功能建模（数据流图），数据建模（实体-联系图），行为建模（状态转换图），框架核心（数据字典）
- 画判断树和判定树

3. 第三章 结构化设计

- 模块独立性：高内聚，低耦合
 - 耦合：程序结构中各个模块之间相互关联的度量
 - 内聚：程序结构中各元素彼此结合的紧密程度
- 体系结构设计：
 - 提高模块独立性，模块规模适中，结构图的深度和宽度适中，结构图中扇入和扇出适中，模块的作用域在控制之内，模块功能的完善化
 - 扇出数：一个模块直接控制的下层模块的个数
 - 扇入数：多个可以有同一个模块，该下属模块的上级模块的个数称为扇入数
- 面对数据流的设计方法：
 - 交换型数据流
 - 事务型数据流
 - 混合型数据流
- 过程设计：
 - 程序流程图
 - N-S 图
 - PAD 图

4. 第四章 结构化编码和测试

- 计算环路复杂度
 - 流图中的区域数等于环路复杂度
 - 流图 G 的环路复杂度 $V(G) = E - N + 2$ ， E 是流图中边的条数， N 是结点数
 - 流图 G 的环路复杂度 $V(G) = P + 1$ ， P 是流图中判定结点的数目

- 软件测试：
 - 黑盒测试：
 - 检测每个功能是否能正常使用，只考虑程序外部结构，不考虑内部逻辑结构，主要针对程序界面和软件功能进行测试
 - 等价类划分法：将程序的输入分为若干类，在每一类中选择少数代表性数据作为测试用例，代表性数据等价于该类中的其他数据
 - 边界值分析法：对等价类划分法的补充，通过选择等价类边界的测试用例进行黑盒测试
 - 错误推断法：基于经验和直觉推测程序中所有可能存在的各类缺陷和错误
 - 因果图法：考虑输入条件，各种输入条件的组合及其产生的输出结果
 - 白盒测试：
 - 把功能模块都测试，保证所有关键路径都被检测到
 - 静态白盒测试：不通过执行程序而进行测试，关键是检查软件的表示和描述是否一致，有无冲突或歧义，包括检查文档内容，界面，代码等
 - 动态白盒测试：结构测试，对程序模块的所有独立的执行路径至少执行一次，对所有逻辑判定执行一次，在循环的边界执行循环体，测试数据内部的有效性
 - 根据覆盖目标不同，逻辑覆盖技术分为：语句覆盖，判定覆盖，条件覆盖，判定-条件覆盖，条件组合覆盖，路径覆盖
- 单元测试：
 - 模块接口测试：通过被测模块接口的数据进行测试
 - 局部数据结构测试
 - 重要路径测试：选择对模块中相对重要的执行路径进行测试
 - 错误处理测试
 - 边界测试

5. 第五章 面对对象分析

- 统一建模语言：UML（事务，关系，图）
 - 关系：依赖关系，关联关系，泛化关系，实现关系
- 对象模型：
 - 对象模型（类图），动态模型（时序模型），功能模型（用例图）
- 面对对象分析：
 - 分析模型：用例模型，对象模型，交互模型
 - 用例图
 - 类图
- 用例之间的关系：包含，拓展，泛化关系

填空题和解答题

填空题

- 软件工程的出现是因为（软件危机）
- 计算机软件由（程序）和（文档）组成
- 传统的软件模型为（瀑布模型）
- 软件工程由方法、工具和（过程）三部分组成
- 结构设计是一种应用最广泛的系统设计方法，是以（数据流）为基础、自顶向下、逐步求精和模块化的过程
- 数据流图有四种基本成分：（数据的源点/终点），数据处理，数据流，数据存储
- 数据流图中的箭头表示（数据流），椭圆或圆形表示（数据处理），矩形表示（数据的源点/终点）
- 需求分析阶段最重要的技术文档是（需求规格说明书）
- （内聚）与（耦合）是模块独立性的两个定性标准，将软件系统划分模块时，尽量做到（高）内聚、（低）耦合，提高模块的（独立性），为设计高质量的软件结构奠定基础
- 软件测试的目标是（确认错误）
- 软件测试的步骤按顺序分别是：（单元测试）、子系统测试、系统测试（子系统和系统测试通常称为集成测试）、（确认测试）和（系统测试（集成测试））
- 软件生命周期一般可分为（问题定义）、可行性研究、（需求分析）、设计编码、（测试）、运行与维护阶段
- 详细设计的工具有（图形工具）、（表格工具）和（语言工具）
- Jackson图除了可以表达程序结构外，还可以表达（数据结构），它首先要分析数据结构，并用适当的工具来描述
- 系统流程图是描绘（物理系统）的传统工具
- （数据流图）和（数据字典）共同构成系统的逻辑模型

解答题

1. 软件生存期各个模型的优点和缺点

- 瀑布模型：
 - 优点：
 - 顺序性和阶段性
 - 可强迫开发人员采用统一规范
 - 缺点：

- 完全依赖文档
 - 可能导致最后的产品不满足用户的需求
 - 适用范围：适用于需求明确和全面，开发过程变化少
- 快速原型模型
 - 优点：
 - 可得到良好需求定义，可适应需求的变化
 - 有利于开发和培训的同步
 - 方便和用户沟通
 - 缺点：
 - 设计困难
 - 缺乏总体质量和可维护性
 - 适用范围：适合大部分场景
- 增量模型：
 - 优点：
 - 人员分配灵活
 - 逐步向用户交付产品，方便用户学习和适应
 - 缺点：
 - 具有不能集成的风险
 - 架构设计要求高，结构开放
 - 适用范围：适用于进行已有产品升级或新版本开发
- 喷泉模型：
 - 优点：
 - 软件开发效率高，开发时间少
 - 各阶段开发没有明显界限，开发人员可以同步开发
 - 缺点：
 - 需要大量开发人员，不利于项目管理
 - 审核要求高
- 螺旋模型：
 - 优点：
 - 设计灵活，可变更
 - 客户参与各阶段开发
 - 缺点：
 - 需要丰富的风险评估经验和专门的知识
 - 增加开发成本，延迟提交时间
 - 使用范围：适合大规模的软件项目
- 敏捷开发模型：
 - 优点：
 - 开发效率高

- 快速响应变化
- 缺点：
 - 不适合大型项目
- 适用范围：适用于小型项目

2. 软件危机的主要原因

- 软件日益复杂和庞大
- 软件开发管理困难和复杂
- 软件开发技术落后
- 生产方式落后
- 开发工具落后
- 软件开发费用不断增加

3. 可行性研究的目的？应该从哪些方面研究目标系统的可行性？

- 目的：
 - 用最小的代价在尽可能短的时间内确定问题是否能够解决
- 可行性：
 - 技术可行性
 - 经济可行性
 - 法律可行性
 - 操作可行性

4. 软件测试包括哪些步骤？说明这些步骤的测试对象是什么？

- 单元测试，测试对象对单元模块
- 集成测试，测试对象为组装后的程序模块
- 确认测试，测试对象为可运行的目标软件系统
- 黑盒测试
- 白盒测试

5. 黑盒测试和白盒测试

- 黑盒测试：
 - 检测每个功能是否能正常使用，只考虑程序外部结构，不考虑内部逻辑结构，主要针对程序界面和软件功能进行测试
 - 等价类划分法：将程序的输入分为若干类，在每一类中选择少数代表性数据作为测试用例，代表性数据等价于该类中的其他数据

- 边界值分析法：对等价类划分法的补充，通过选择等价类边界的测试用例进行黑盒测试
- 错误推断法：基于经验和直觉推测程序中所有可能存在的各类缺陷和错误
- 因果图法：考虑输入条件，各种输入条件的组合及其产生的输出结果
- 白盒测试：
 - 把功能模块都测试，保证所有关键路径都被检测到
 - 静态白盒测试：不通过执行程序而进行测试，关键是检查软件的表示和描述是否一致，有无冲突或歧义，包括检查文档内容，界面，代码等
 - 动态白盒测试：结构测试，对程序模块的所有独立的执行路径至少执行一次，对所有逻辑判定执行一次，在循环的边界执行循环体，测试数据内部的有效性
 - 根据覆盖目标不同，逻辑覆盖技术分为：语句覆盖，判定覆盖，条件覆盖，判定-条件覆盖，条件组合覆盖，路径覆盖

6. 软件设计的过程

- 软件设计是把许多事物和问题抽象起来，并且抽象它们不同层次和角度，是将需求转变为软件陈述的过程，是迭代的过程
- 软件设计，要根据需求规格说明书，对整个设计过程进行计划，然后实施具体的设计过程，即“先整体，再局部”，也是不断迭代和精化的过程
- 然后需要对生成的设计规格说明书进行评审，启动 质量评价的标准，若未通过评审，需重新修改设计，直至评审通过，确定最后定型的过程本身。进入后续阶段，完成软件设计过程

7. 衡量模块独立性的两个定性标准？定义是？

- 模块独立性：高内聚，低耦合
 - 耦合：程序结构中各个模块之间相互关联的度量
 - 内聚：程序结构中各元素彼此结合的紧密程度

8. 软件测试的基本任务是什么？测试与调试的区别是什么？

- 任务是在软件投入生产性运行之前，尽可能多地发现软件中的错误
- 与测试的区别：测试是为了发现软件中的错误，而调试是诊断并改正错误

9. 软件测试有几种方法？每种方法的特点是什么

- 单元测试：在模块接口、局部数据结构、重要的执行通路、出错处理通路、边界条件5个方面来对模块进行测试
- 集成测试：是测试和组装软件的系统化技术可分为自顶向下和自底向上两种集成策略
- 确认测试：验证软件的有效性
- 白盒测试：也称结构测试，检验程序中的每条通路是否都能按预定要求正确工作

- 黑盒测试：也称功能测试，检测是否每个功能都能正常使用

10. 简述面向对象的4个要点的含义

面向对象的 4 个要点是：对象分解、数据专有、继承、封装性

- 对象分解：认为客观世界是由各种对象组成的
- 数据专有，方法共享：把所有对象都划分成各种对象类(简称为类，class)，每个对象类都定义了一组数据和一组方法
- 继承：按照子类(或称为派生类)与父类(或称为基类)的关系，把若干个对象类组成一个层次结构的系统(也称为类等级)
- 封装性：对象彼此之间仅能通过传递消息互相联系