

题目总结

选择

- 软件生命周期：问题定义、可行性研究、需求分析、总体设计、详细设计、编码和单元测试、综合测试、运行维护
- 软件生命周期的各个阶段中，工作量最大的阶段是（软件维护）
- 瀑布模型的特点不包括（进入某个阶段工作后，不再回复到之前阶段的工作）
- 需求分析阶段的工作不包括（建立目标系统的物理模型）
- 总体设计阶段的工作不包括（确定实现各个模块功能的处理逻辑）
- 描绘系统物理模型的传统工具是（系统流程图）
- 符合信息隐藏原理的是（将可能要修改的设计决策隐藏起来）
- 有利于提高模块独立性的做法是（使判定作用范围内的模块尽量成为该判定所在模块的直属下级模块）
- 有关结构化设计方法的正确叙述是（是一种面向数据流的设计方法）
- 有关总体设计阶段所使用的结构图的不正确叙述是（结构图种的模块是按照自上而下、自左向右的顺序执行的）
- 要求使用顺序、选择和循环控制结构的组合或嵌套来表达程序的过程设计工具是（盒图）
- 关于好的编码风格的正确叙述是（检查输入项重要组合的合法性）
- 能发现软件需求规格说明书种的错误的测试步骤是（验收测试）
- 自顶向下和自底向上集成测试都具有的优点是（支持故障隔离）
- 不符合面向对象设计准则的是（尽可能减小继承耦合度）
- 软件工程的三要素：过程、方法、工具。
- 通常使用（软件生命周期模型）描述软件过程，它规定了（生命周期阶段划分）和（各阶段执行顺序）
- 需求分析阶段的任务包括确定对系统的综合要求和分析系统的（数据需求），综合要求包括（功能需求）、（性能需求）、可靠性和可用性需求、出错处理需求、约束、接口需求、逆向需求和将来可能提出的需求。
- 对系统提出一组要求后，一般从下述4个方面验证这些需求的正确性：（一致性）、（完整性）、现实性和有效性。
- 软件测试的基本方法有白盒测试和黑盒测试法。单元测试主要使用（白盒法），验收测试主要使用（黑盒法）。
- 白盒测试中，对源程序语句检测详尽程度最高的逻辑覆盖标准是（条件组合覆盖），对程序路径覆盖程度最高的逻辑覆盖标准是（路径覆盖）。
- 使用黑盒测试法时，总是联合使用（等价分类）和（边缘值分析）两种技术。
- 面向对象方法开发软件通常需要建立三种模型：（对象模型-类图）、（动态模型-状态图）和功能模型-用例图。
- 功能模型可以用用例图来表示，用例图中的用力对应于（对象模型）中的类所提供的服务，用例可能产生（动态模型）中的一系列事件，这些事件的产生可以用（顺序图）来表示。
- RUP把软件生命周期分成哪4个阶段：初始阶段、精化阶段、构建阶段、移交阶段。
- 软件工程的本质特性：
 1. 关注于大型程序的构造。
 2. 中心课题是控制复杂性。
 3. 软件经常变化。

4. 开发软件的效率非常重要。
5. 和谐的合作是开发软件的关键。
6. 软件必须有效地支持它的用户。
7. 在软件工程领域中，通常由具有文化背景的人替具有另一种文化背景的人创造产品。

近

- （软件本身特点）不是造成软件危机的主要原因。
- 通常最终交付的产品可能不是用户真正需要的模型是（瀑布模型）
- （逻辑覆盖）不属于黑盒测试技术
- 集成测试所依据的模块说明书和测试方案应在（详细设计）阶段完成，它能发现设计错误
- （继承）不仅增加了面向对象软件系统的灵活性，进一步减少了信息冗余，还提高了软件的可重用性和可扩展性。
- 面向对象技术中，（本质上具有并行性）是对象的基本特点。
- （可理解性、可重用性，可测试性）是决定软件可维护性的基本因素。
- （提供严格定义的文档）不是原型化方法所具有的特征。
- （改善性维护）是占比最多的
- （顺序内聚）不是面向对象设计中存在的内聚。
- （用例图）指明了系统应该做什么
- 面向对象方法的一个主要目标是要提高软件的（可重用性）。
- 喷泉模型描述的是面向对象的开发过程，反映了该开发过程的（迭代和无间隙）。
- 为了解决这个问题，目标系统必须做什么对应（需求分析）
- （继承重用）不是软件成分的重用
- 从结构化的瀑布模型看，在生命周期中，（需求分析）阶段环节出错，对软件的影响最大。
- 以下选项不是软件工程的本质特征的是（结果应该能清楚地审查）
- RUP把软件生命周期划分成4个阶段：初始阶段、细化阶段、构建阶段、移交阶段，没有（维护阶段）。

判断

- [F] 对象可以按其属性来归类，子类可以通过泛化机制获得其父类的特征。
- [F] 从一张顺序图可以得到其所涉及的各个对象的完整状态图。
- [T] 软件维护阶段的大部分工作是增加新功能、修改已有功能或对系统进行改进。
- [T] 在图形类过程设计工具中，使用PAD描述的处理过程最容易转换成与之对应的高级语言程序。
- [T] 极限编程具有对变化和不确定性的更快速，更敏捷的反应特性。
- [F] 风险驱动也可能是螺旋模型的缺点
- [F] 自顶而下测试方法的主要优点是不需要存根程序，但需要测试驱动程序。（反了）
- [T] 面向对象设计中，对象设计确定解空间中的类，关联，接口形式及实现服务的算法。
- [T] 软件本质上的复杂性使软件产品难以理解，影响软件产品的可靠性，并使维护过程变得困难。

[T] 模块独立的概念是模块化、抽象、信息隐藏和局部化概念的直接结果。