

复习提纲总结

一、综合题目

- 1、能分析可行性
- 2、画出数据流图
- 3、映软件结构
- 4、工程网络
- 5、根据算法画出盒图， PAD 图
- 6、用 Jackson 图表示数据结构
- 7、根据题意，画出对象模型
- 8、类似142页第8题
- 9、用有穷状态机进行规格说明
- 10、能用不同的方法预测错误总数

二、简答题

1、广义的说，软件重用可以分为哪几个层次？

1.重用

重用也叫再用或复用,是指同一事物不作修改或稍加改动就多次重复使用，广义地说,软件重用可分为以下3个层次：

- (1)知识重用(例如,软件工程知识的重用)
- (2)方法和的标准的重用(例如,面向对象方法或国家制定的软件开发规范的重用)
- (3)软件成分的重用

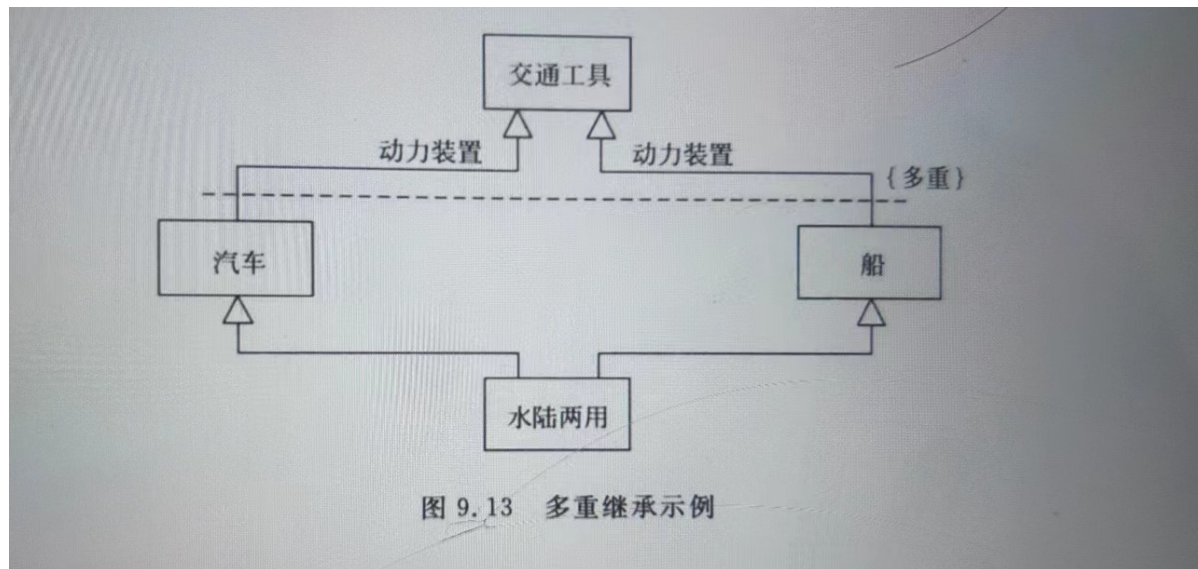
2、软件过程的输出信息可以分为哪几类？

软件过程的输出信息可以分成三个主要类别：

- (1) 计算机程序（源代码和可执行程序）；
- (2) 描述计算机程序的文档（针对不同的软件开发人员和客户）;
- (3) 数据或内容（包含在程序内部和外部）。在软件过程中产生的所有信息项统称为软件配置。

3. 什么是多重继承？

承多重继承指的是，一个子类可以同时多次继承同一个上层基类，例如图9.13中的水陆两用类继承了两次交通工具类。



4. 什么是软件工程？

- 软件工程是从管理和技术两方面研究如何更好地开发和维护计算机软件的一门新兴学科，采用工程概念、原理、技术和方法来开发和维护软件，用正确的管理技术和当前最好技术方法结合，以经济地开发高质量软件并有效维护。

5. 什么是软件危机？如何解决？

1. 什么是软件危机

- 软件危机是指在计算机软件的开发和维护过程中所遇到的一系列严重问题。
- 软件危机包含两方面的问题：
 - ①如何开发软件，以满足对软件日益增长的需求；
 - ②如何维护数量不断膨胀的已有软件。

2. 解决:

1. 应该对计算机软件有一个正确的认识。
2. 应该充分认识到软件开发是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目
3. 应该推广使用在实践中总结出来的开发软件的成功的技术和方法，并且研究探索更好更有效的技术和方法，尽快消除在计算机系统早期发展阶段形成的一些错误概念和做法。
4. 应该开发和使用更好的软件工具

6. 一般来说，至少应该从哪些方面研究每种解法的可行性？

- (1)技术可行性：工使用现有的技术能实现这个系统吗？
- (2)经济可行性：这个系统的经济效益能超过它的开发成本吗？
- (3)操作可行性：系统的操作方式在这个用户组织内行得通吗？

7. 一般来说, 应该从哪4个方面验证需求的正确性?

需求分析阶段的工作结果是开发件系统的重要基础

- (1)一致性 所有需求必须是一致的,任何一条需求不能和其他需求互相矛盾。
- (2)完整性 需求必须是完整的,规格说明书应该包括用户需要的每一个功能或性能
- (3)现实性 指定的需求应该用现有的硬件技术和软件技术基本上可以实现的。对硬件技术的进步可以做些预测,对软件技术的进步则很难做出预测,只能从现有技术水平出发判断需求的现实性。
- (4)有效性 必须证明需求是正确有数的,确实能解决用户面对的问题

8. 什么是形式化方法?

软件形式化方法是指建立在严格数学基础上的软件开发方法。在逻辑科学中是指分析、研究思维形式结构的方法。它把各种具有不同内容的思维形式(主要是命题和推理)加以比较,找出其中各个部分相互联结的方式,如命题中包含概念彼此间的联结,推理中则是各个命题之间的联结,抽取出它们共同的形式结构;再引入表达形式结构的符号语言,用符号与符号之间的联系表达命题或推理的形式结构。

9. 应该用正式的文档记录总体设计的结果, 在这个阶段应该完成的文档通常有哪几种?

书写文档阶段: 应该用正式的文档记录总体设计的结果, 在这个阶段应该完成的文档通常有下述几种: 系统说明、用户手册、测试计划、详细的实现计划以及数据库设计结果。

10. 代码审查比计算机测试的优越性表现在哪里?

代码审查比计算机测试优越的是: 一次审查会上可以发现许多错误; 用计算机测试的方法发现错误之后, 通常需要先改正这个错误才能继续测试, 因此错误是一个一个地发现并改正的。也就是说, 采用代码审查的方法可以减少系统验证的总工作量。

11. 有那4类维护活动?

1. 改正性维护: 诊断改正使用过程中发现的软件错误
2. 适应性维护: 修改软件以适应环境变化
3. 完善性维护: 根据用户要求改进或扩充软件使它完善
4. 预防性维护: 修改软件, 为将来的维护活动预先做准备

12. 面向对象方法学有哪些优点?

④与传统方法学的对比

- a.传统方法学强调自顶向下顺序地完成软件开发的各阶段任务。
- b.用面向对象方法学开发软件的过程, 是一个主动地多次反复迭代的演化过程。面向对象方法在概示方法上的一致性, 保证了在各项开发活动之间的平滑(即无缝)过渡。

⑤优点

- a.降低了软件产品的复杂性;
- b.提高了软件的可理解性;
- c.简化了软件的开发和维护工作;
- d.提高了软件的可重用性。

13. 类与类之间通常有哪几种关系？

类与类之间可以有以下几种关系：

- 1.继承关系 (Inheritance)：表示一个类是另一个类的子类，继承了父类的属性和方法。
- 2.实现关系(Implementation)：表示一个类实现了一个接口，必须实现接口中定义的所有方法。
- 3.关联关系(Association)：表示类之间的拥有关系，一个类可以拥有另一个类的对象作为其属性。
- 4.聚合关系(Aggregation)：表示整体和部分之间的关系，一个类可以拥有另一个类的对象作为其部分。
- 5.组合关系(Composition)：表示一种强的聚合关系，整体和部分的生命周期是一致的。
- 6.依赖关系 (Dependency)：表示一个类依赖于另一个类，当一个类的方法需要另一个类的对象作为参数或者返回值时，就会出现依赖关系。

14. 大型系统的对象模型通常由哪几个层次组成？

- 1.主题层
- 2.类与对象层
- 3.结构层
- 4.属性层
- 5.服务层

15. 测试单个类的方法主要有哪些？

1. [单元测试](#)：单元测试是对软件中的最小可测试单元进行检查和验证。对于面向对象编程，这最小的单元就是方法，即类中的单个方法。
2. [集成测试](#)：在单元测试的基础上，将所有模块分组，测试组合后的模块。这种方法也称为[增量测试](#)，它测试的是组合后的模块，在通过单元测试的模块中逐步增加，直到整个程序可以通过为止。
3. [系统测试](#)：此时，软件作为系统的一部分，与硬件、网络等其他系统元素组合在一起进行测试。重点测试整个系统的功能和性能。
4. [回归测试](#)：在修改或重构软件后进行测试，以确保修改没有引入新的错误。
5. [冒烟测试](#)：对需求或功能的某个子集进行快速测试，以判断是否可以开始更全面的测试。
6. [黑盒测试](#)：也称为功能测试，主要关注软件的功能和需求是否满足用户期望，不考虑代码结构和内部逻辑。
7. [白盒测试](#)：也称为结构测试、透明盒测试、逻辑驱动测试或基于代码的测试，它是按照程序内部的结构[测试程序](#)，简单来说就是代码层面的测试。白盒测试在路径覆盖上一定程度是可以做到穷尽测试的。
8. [灰盒测试](#)：介于黑盒测试和白盒测试之间，它关注的是输出对于输入的正确性，同时也关注内部处理过程，但并不关注内部结构和实现方式。

16. 能力成熟度的5个等级是什么？

软件能力成熟模型CMM (Capability Maturity [Model](#) for Software) 的五个成熟级别

第一级：初始级(Initial)；**无序，随意**

初始级的软件过程是无序的，项目的执行是随意甚至是混乱的。工作方式处于救火状态，不断的应对突如其来的危机；

第二级：可重复级(Repeatable)；**基本的项目管理**

建立了基本的项目管理过程来跟踪费用，进度和功能特性，制定了必要的过程纪律，能重复早先类似的应用项目取得的成功。

第三级：已定义级(Defined)；**标准化，文档化**

已经将软件管理工程两方面的过程文档化，标准化，并综合成组织的标准软件过程，所有项目均使用该标准开发维护软件。

第四级：已管理级(Managed)；可预测

收集对软件过程和产品质量的详细度量，对软件过程和产品都有定量的理解和控制。

第五级：优化级(Optimizing)。优化

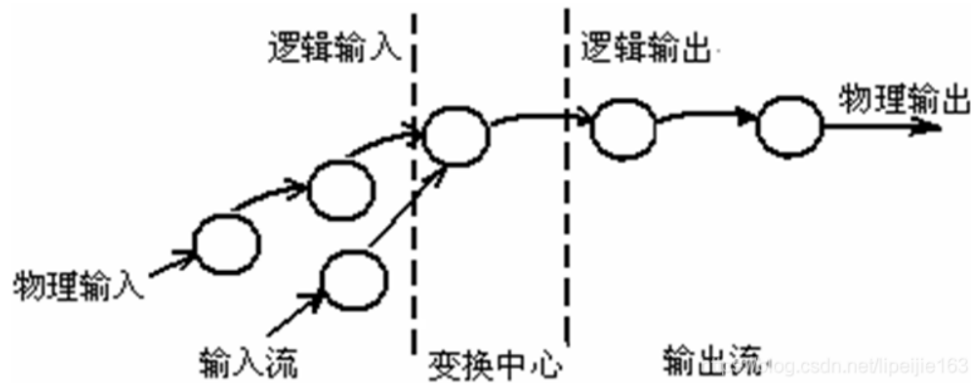
过程的量化反馈和先进的新思想，新技术促使过程不断改进。

17. 用图表示事物流与变换流。

1. 变换型数据流图

根据信息系统的模型，信息一般是以外部形式进入系统，通过系统处理后，然后离开系统。从其过程可以得出，变换流的数据流图是一个线性结构。

变换型的数据流是由输入、变换（或称处理）和输出三部分组成。



2. 事务型数据流图

若某个加工将它的输入流分离成许多发散的数据流，形成许多平行的加工路径，并根据输入的值选择其中一个路径来执行，这种特征的数据流图称为事务型数据流图。

图中的处理T称为事务中心，它完成下述任务：

- (1) 接收输入数据。
- (2) 分析每个事务，确定其类型。
- (3) 根据事务选择一条活动通路。

