

2022 年软件工程复习提纲

一、选择题（本题共 10 小题，每小题 2 分，共 20 分）

1. 下面所列的性质中， D 不属于面向对象程序设计的特性。

（A）继承性 （B）重用性 （C）封装性 （D）可视化

二、简答题（本题共 4 小题，共 20 分）

1. 软件工程的定义

软件工程是指导计算机软件开发和维护的一门工程学科，采用工程的概念、原理、技术和方法对软件进行开发和维护。把经过时间检验的管理技术和能得到的最好的技术方法结合起来，以经济地开发出高质量的软件并有效的进行维护。

2. 软件生命周期（3 个时期，8 个阶段）

软件生命周期分为问题定义、可行性研究、需求分析、开发阶段、维护这 5 个阶段。各个阶段的主要任务是如下：

1、问题定义

要求系统分析员与用户进行交流，弄清“用户需要计算机解决什么问题”然后提出关于“系统目标与范围的说明”，提交用户审查和确认。

2、可行性研究

一方面在于把待开发的系统的目标以明确的语言描述出来，另一方面从经济、技术、法律等多方面进行可行性分析。

3. 总体设计

4. 详细设计

5. 编码与单元测试

3、需求分析

弄清用户对软件系统的全部需求，编写需求规格说明书和初步的用户手册，提交评审。

4、开发阶段

开发阶段由四个阶段组成：概要设计、详细设计、实现、测试

5、维护

维护包括四个方面：

（1）改正性维护：在软件交付使用后，由于开发测试时的不彻底、不完全、必然会有一部分隐藏的错误被带到运行阶段，这些隐藏的错误在某些特定的使用环境下就会暴露。

（2）适应性维护：是为适应环境的变化而修改软件的活动。

（3）完善性维护：是根据用户在使用过程中提出的一些建设性意见而进行的维护活动。

（4）预防性维护：是为了进一步改善软件系统的可维护性和可靠性，并为以后的改进奠定基础。

3. 软件开发模型（瀑布模型、喷泉模型、增量、螺旋、快速原型）

瀑布模型：

1.里程碑或基线驱动。2.过程逆转性很差或者说不可逆转。逆转可能会延误工期，增加成本，造成损失

1.开发阶段清晰，便于评审、审计、跟踪、管理和控制。

1.不可逆或很难可逆。2.问题会积累，错误会传递发散扩大，导致成本和质量失控。

1.在开发时间内需求不变化或很少变化。2.分析设计人员对此领域非常熟悉。3.低风险项目。

4.用户使用环境稳定（如系统软件，工具软件）

快速原型模型

1.容易适应需求的变化

1.克服瀑布模型的缺点，减少由于软件需求不明确带来的开发风险。

1.所使用的开发工具和技术不一定符合主流的发展；快速建立起来的系统架构加上连续的修改可能会导致产品质量低下。

1.需求不明确或复杂系统。2.用户无法自主提出应用需求。

增量模型：

1.软件由一系列增量构件组成

1.人员分配灵活，刚开始不用投入大量人力资源。如果核心产品很受欢迎，则可增加人力实现下一个增量。当配备人员不能在设定的时间内完成产品时，它提供了一种先推进核心产品的途径，这样即可先发部分功能给客户，对客户起到镇静剂作用。2.增量能够有计划的管理技术风险。

1.由于各个构件是逐渐并入已有的软件体系结构中，所以加入构件必须不破坏已构造好的系统部分，这需要软件具备开放式的体系结构。2.在开发过程中，需求变化是不可避免的，增量的灵活性可以使其适应这种变化的能力大大优于瀑布和快速原型模型，但也容易退化为边改边做模型，从而使软件过程的控制失去整体性。3.如果增量包之间存在相交的情况且未很好处理，则必须做全盘系统分析，这种模型将功能细化后分别开发的方法较适应于需求经常改变的软件开发过程

1.系统容易拆分。2.开发人力比较少。3.特别适用于商业软件（如QQ，网游）

螺旋模型：

1.是一种周期性的方法进行系统开发。2.有许多“中间版本”。3.每个周期都包括需求定义、风险分析、工程实现和评审4个阶段

1.设计上灵活，可在项目各个阶段进行变更。2.以小的分段来构建大型系统，使成本计算变得简单容易。3.客户始终参与每个阶段的开发，保证了项目不偏离正确方向以及项目的可控性。

1.建设周期长，而软件技术更新比较快，所以经常出现软件开发结束后，和当前的技术水平有了很大的差距，无法满足当前用户需求。

1.系统庞大，风险高。2.需求不太明确。

喷泉模型：

1.该模型的各个阶段没有明显的界限，开发人员可以同步进行开发。

1.提高软件项目开发效率，节省开发时间，适用于面向对象的软件开发过程。

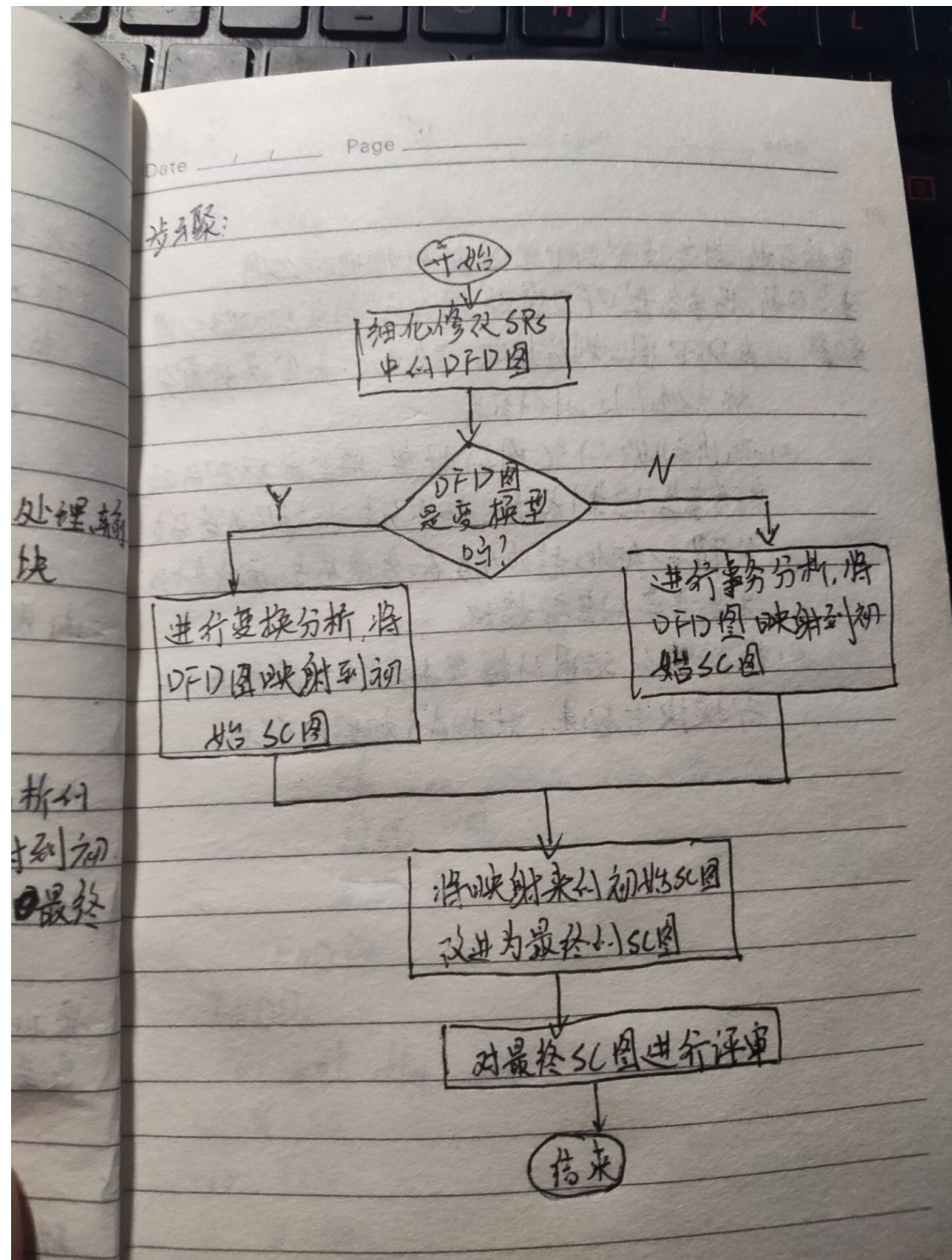
1.由于喷泉模型在各个开发阶段是重叠的，因此在开发过程中需要大量的开发人员，因此不利于项目的管理。2.要求严格管理文档，使得审核难度加大，尤其是面对可能随时加入的各种信息、需求与资料的情况。

1.面向对象的软件开发过程

4.结构化分析的目的、任务和步骤

答：定义：结构化分析实质上是一种创建模型的活动。为了开发出复杂的软件系统，系统分析员应该从不同角度抽象出目标系统的特性，使用精确的表示方法构造系统的模型，验证模型是否满足用户对目标系统的需求，并在设计过程中逐渐把和实现有关的细节加进模型中，直至最终用程序实现模型。

任务：结构化设计包括架构设计、接口设计、数据设计和过程设计等任务。



5.需求分析阶段的图解工具（DFD、DD、ER图、IPO图、层次方框图、Warnier图）

6. 软件工程的基本定理

基本原理：

1. 用分阶段生命周期计划严格要求。
2. 坚持进行阶段评审。
3. 实行严格的产品控制。
4. 采用现代程序设计技术。
5. 结果应能清楚地审查。
6. 开发小组的成员应该少而精。
7. 承认不断改进软件工程实践的必要性。

7. 衡量模块独立性的指标（内聚、耦合）

答：内聚和耦合

模块的独立程度可以由两个定性标准衡量，这两个标准分别是内聚和耦合。

耦合衡量不同模块彼此间互相依赖（连接）的紧密程度；内聚衡量一个模块内部各个元素彼此结合的紧密程度。

8. 总体设计采用的图解工具（HC 图、SC 图）

1. 层次图：描述层次结构。
2. HIPO 图：在层次图的基础上，把图中除了顶层的方框外都加上编号****基本形式：输入，处理，输出。
3. 结构图（SC）：表达程序结构图形的表示方法，反映程序模块间的层次关系和联系。
成分：模块，模块间调用关系，通信，辅助控制符号。
结构图的四中类型：传入，传出，变换，协调

9. 结构化设计的目的、任务和步骤

答：结构化设计的目的：使程序的结构尽可能反映要解决的问题的结构。

结构化设计的任务：把需求分析得到的数据流图 DFD 等变换为系统结构图（SC）。

基本步骤：分为概要设计和详细设计两个阶段。

10. 变换分析和事务分析

变换分析设计是一个顺序结构，由输入、变换和输出三部分组成，其工作过程有 3 步：取得数据、变换数据和给出数据。

事务分析设计是将它的输入流分离成许多发散的数据流，形成许多加工路径，并根据输入的值选择其中一个路径来执行。

变换分析主要是将变换型的 DFD 图映射为与之相应的初始 SC 图。采用以下几步来完成：

- （1）在 DFD 图上区分输入、变换中心和输出 3 个部分，标出它们之间的分界。
- （2）画出初始的 SC 图的框架，这一框架由顶层的主控模块和供它调用的输入、变换处理和

输出三个分支框组成。

(3) 对输入、变换处理和输出这三个分支进行自顶向下的分解，画出各个分支所需要的全部模块。

(4) 将初始的 SC 图的框架和各个分支所需要的全部模块合起来，就构成了初始的 SC 图。事务分析主要是将事务型的 DFD 图映射为与之相应的初始 SC 图。采用以下几步来完成：

(1) 在 DFD 图上划分接收、事务中心和发送 3 个部分，标出它们之间的分界。

(2) 画出初始的 SC 图的框架，将上述 3 个部分映射为事务控制模块、接收分支和动作发送分支。

(3) 分解和细化接收分支和发送分支，画出各个分支所需要的全部模块。

(4) 将初始的 SC 图的框架和各个分支所需要的全部模块合起来，就构成了初始的 SC 图。

11. 软件测试的目的和步骤

测试过程按 4 个步骤进行，即单元测试、集成测试、系统测试及验收测试。

单元测试和编码对应，集成测试和详细设计对应，系统测试和概要设计对应，验收测试和需求分析对应。

12. 黑盒测试和白盒测试，分别采用的技术

如果产品内部活动方式已经知道，可以测试它的内部活动是否符合设计要求。将被测程序视为一个透明的盒子，根据被测程序的内部结构来设计测试用例。

白盒测试采用逻辑覆盖测试法和路径测试法，逻辑覆盖有 5 种覆盖标准：

- ① 语句覆盖
- ② 判定覆盖
- ③ 条件覆盖
- ④ 判定-条件覆盖
- ⑤ 条件组合覆盖

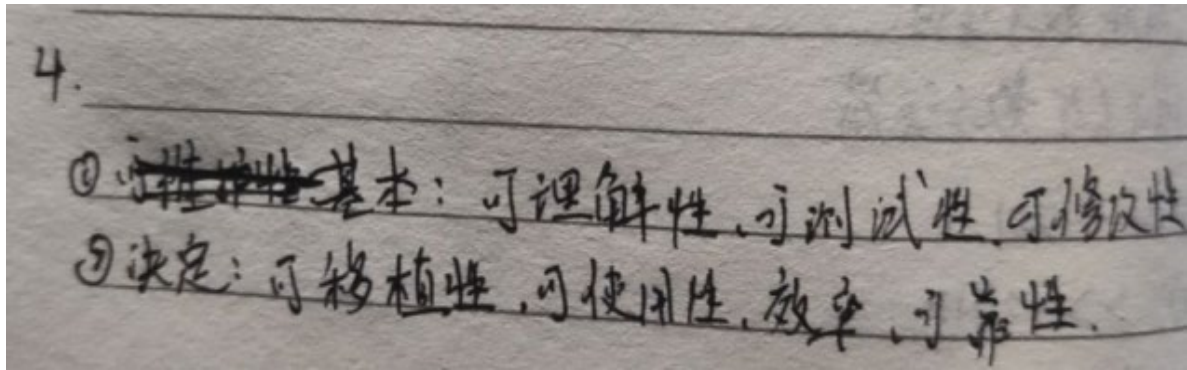
答：黑盒测试又称为“功能测试”，是将测试对象看做一个黑盒，在并不考虑软件产品的内部结构和处理过程的基础上对软件产品进行功能测试。黑盒测试注重软件产品的“功能性需求”。

黑盒测试常用的技术有：边界值、等价类、错误推测法、功能分解法、因果图、判定表、正交试验法、场景法。

13. 软件 4 种维护活动

答：错性维护（校正性维护）、适应性维护、完善性维护或增强、预防性维护或再工程。

14. 可维护性基本因素、决定因素



15.面向对象方法学定义、优点

答: 面向对象方法学的出发点和基本原则, 是尽可能模拟人类习惯的思维方式, 使开发软件的方法与过程尽可能接近人类认识世界解决问题的方法与过程, 也就是使描述问题的问题空间(也称为问题域)与实现解法的解空间(也称为求解域)在结构上尽可能一致。

优点: (1)与人类习惯的思维方法一致(2)稳定性好(3)可重用性好(4)可维护性好

16.对象模型、动态模型和功能模型分别采用的图解的工具

答: 对象模型表示静态的、结构化的系统的“数据”性质。它是对模拟客观世界实体的对象以及对象彼此间的关系的映射, 描述了系统的静态结构。

动态模型表示瞬时的、行为化的系统的“控制”性质, 它规定了对象模型中的对象的合法变化序列。

功能模型表示变化的系统的“功能”性质, 它指明了系统应该“做什么”, 因此更直接地反映了用户对目标系统的需求。

3 种模型之间的关系如下:

- (1) 针对每个类建立的动态模型, 描述了类实例的生命周期或运行周期。
- (2) 动态模型中的状态转换驱使行为发生, 这些行为在数据流图中被映射成处理, 同时与对象模型中的服务相对应。
- (3) 功能模型中的处理对应于对象模型中的类所提供的服务。通常, 复杂的处理对应于复杂对象提供的服务, 简单的处理对应于更基本的对象提供的服务。一个处理可能对应多个服务, 一个服务也可能对应多个处理。
- (4) 功能模型(数据流图)中的数据存储, 以及数据的源点/终点, 通常是对象模型中的对象。功能模型中的数据流, 往往是对象模型中对象的属性值, 也可能是整个对象。
- (5) 功能模型中的处理(或用例)可能产生动态模型中的事件。

三. 画图题 (本题共 2 小题, 共 20 分)

1. 流程图、N-S 图、PAD 图
2. 瀑布模型、喷泉模型

四. 改错题 (本题共 1 小题, 共 10 分)

请将下面的非结构化程序改成结构化的程序。

作业例子

五. 综合题（本题共 3 小题，共 30 分）

1. 判定树、判定表
2. 用黑盒测试法设计测试用例
3. UML 中的用例图