

软件工程

一. 软件工程重点知识

1. 单句整理

1) 一部分

- ① 从结构化的瀑布模型看，在它的生命周期中的各阶段中，需求分析阶段出错，对软件的影响最大。
- ② 螺旋模型将瀑布模型和演化模型相结合，并增加了风险分析。
- ③ 结构化分析方法是一种预先严格定义需求的方法，它在实施时强调的是分析对象的数据流。
- ④ 进行需求分析可以使用数据流图（DFD），判定表，数据字典。
- ⑤ 在面向数据流的软件设计方法中，一般将信息流分为变换流和事务流。
- ⑥ PAD图的控制执行流程为自上而下，从左到右。
- ⑦ 概要设计阶段产生的文档包括概要设计说明书，数据库设计说明书，用户手册。
- ⑧ 与选择编程语言有关的因素是软件开发的方法，软件执行的环境，软件开发人员知识。
- ⑨ 软件工程针对维护工作的主要目标是提高软件的可维护性，降低维护的代价。
- ⑩ 开发软件工具的主要目的是为了提高软件生产率和改善软件质量。
- ⑪ 在软件需求分析中，必须采用合理的步骤，才能准确地获取软件的需求，可归纳为4个步骤：需求获取，分析建模，文档编写，需求验证。
- ⑫ 耦合衡量不同模块彼此间互相依赖（连接）的紧密程度，模块间的耦合性越高，其模块的独立性就越弱。
- ⑬ 在面向对象的设计方法中，对象和传递消息分别是表现事物及事物间相互联系的概念。
- ⑭ 用等价类划分的方法设计测试用例的步骤为，划分等价类和确定测试用例。

2) 二部分

- ① 软件开发环境支持一定的软件开发方法，或按照一定的软件开发模型组织而成。
- ② 技术可行性要解决技术风险问题。
- ③ 在软件需求规范中，执行要求可以归类为过程要求。
- ④ 在分层数据流图映射成软件结构的设计中，变换型通常用于高层数据流图的转换，而事务型通常用于较低层数据流图的转换这种说法是错误的。
- ⑤ JACKSON方法根据数据间的控制结构来导出程序结构。
- ⑥ 对象是面向对象开发方法的基本成分，每个对象可用它本身的一组属性和它可执行的一组操作来定义。
- ⑦ UML语言支持的建模方式有动态建模，功能建模，静态建模。
- ⑧ 面向对象程序设计的基本机制包括继承，消息，方法。
- ⑨ 软件的可维护性变量可分解为对多种因素的度量，可测试性，可理解性，可修改性是可维护度量内容。
- ⑩ 软件工具的发展特点是软件工具由单个工具向多个工具集成化方向发展，重视用户界面的设计，不断的采用新理论和新技术。
- ⑪ 描述软件开发过程中各种活动如何执行的模型称为周期模型。
- ⑫ 可行性研究具体步骤的最后一步是编写可行性报告。
- ⑬ 模块间还经常用带注释的端箭头来表示模块调用过程中来回传递的信息，有时箭头

尾部带空心圆表示传递的是数据，带实心圆的标识传递的是控制信息。

- ⑭ 如果一个特殊类直接地继承两个以上的一般类，则称为多继承。

3) 三部分

- ① 软件工程学的目的是最终解决软件生产的使软件生产工程化问题。
- ② 软件工程方法中普遍应用的方法之一是结构化生命周期方法（SLC 方法），即严格定义需求，规范文档格式，划分开发阶段。
- ③ 研究开发资源的有效性是进行技术可行性研究的一方面。
- ④ 软件需求分析阶段的工作，可以分成以下 4 个方面，对问题的识别，分析与综合，指定规格说明以及需求分析评审。
- ⑤ 模块内的某成分的输出是另一成分的输入，该模块的内聚度是顺序性的。
- ⑥ JACKSON 方法主要适用于规模适中的数据处理的系统的开发。
- ⑦ 面向对象的主要特征除对象唯一性，封装，继承外，还有多态性。
- ⑧ 为了提高易读性，源程序内部应加功能性注释，用于说明程序段或语句的功能。
- ⑨ 从已经发现故障的存在到找到准确的故障位置并确定故障的性质，这一过程称为调试。
- ⑩ 人们将在软件运行/维护阶段对软件产品所进行的修改称为“维护”，校正性维护是由于开发时测试的不彻底，不完全造成的。
- ⑪ 结构化方法总的思想是自顶向下逐步求精，它的基本原则是功能的分解和抽象，
- ⑫ 成本-效益分析首先是估算将要开发的系统的开发成本，然后与可能取得的效益进行比较和权衡。
- ⑬ 需求分析阶段所要编写的文档有，需求规格说明书，初步用户使用手册，确认测试计划。
- ⑭ 模块间耦合高低取决于模块间接口的复杂性，调用的方式及传递的信息。
- ⑮ 必须在软件交付之前对整个软件配置进行评审，以减少文档的副作用。

4) 四部分

- ① 软件开发时，一个错误发现的越晚，为改正它们付出的代价越大。
- ② 软件工程学涉及到软件开发技术和工程管理两方面内容，包括软件开发方法，软件开发工具，软件工程环境。
- ③ 数据流图（DFD）是软件开发需求分析阶段经常使用的工具。
- ④ 模块内部联系最大的是功能内聚。
- ⑤ 软件需求分析的内容是确定开发的软件应有哪些功能，应达到什么性能。
- ⑥ 软件设计中的概要设计过程其主要描述的是模块间关系。
- ⑦ 软件测试的目的是发现软件的错误。
- ⑧ 与测试数据有关的文档是该软件的设计人员，程序的复杂度，源程序。
- ⑨ 数据耦合，控制耦合，公共耦合耦合性递增。
- ⑩ 在软件开发过程中，编程语言的选择很重要，如果一个项目侧重于数据结构设计，则应选择 C 语言，
- ⑪ 用 JACKSON 图既可表示数据结构，又可以表示程序结构。
- ⑫ 面向对象的程序设计语言具有数据抽象，信息隐蔽，信息继承。
- ⑬ 软件维护工作中大部分工作是由于用户需求的改变而引起的。
- ⑭ 软件的面向数据流的设计方法，利用其定义的映射方式可以把数据流图变换成软件结构，在映射中，一般将数据流分为变换流和事务流两种。
- ⑮ 程序的三种基本控制结构是顺序结构，分支结构，循环结构，他们的共同点是单入口单出口。

5) 五部分

- ① 软件生命周期中所花费用最多的阶段是**软件维护**。
- ② 用于表示模块间调用关系的图叫**SC**。
- ③ 软件是一种**逻辑**产品。
- ④ SD 方法的最终目标是**块间联系小，块内联系大**。
- ⑤ 保证软件质量的手段有复审，复查，管理复审和测试等，其中复审发生在软件生命周期的**每个阶段结束之前**。
- ⑥ 面向对象程序设计语言不同于其他语言的最主要特点是**继承性**。
- ⑦ 提高软件质量和可靠性的技术大致分为两大类，一类是避开错误技术，另一类是**容错技术**。
- ⑧ 一个只有顺序结构的程序，其环形复杂度是**0**。
- ⑨ 按照软件过程的系统方法进行软件的工程活动和管理活动，进而不断完善软件各个软件过程，从而不断提高**软件过程能力**。
- ⑩ 模块越小，优点越明显，一般来说，模块大小都在 10 行以下。
这是**错误**的，10 行以下太过极端。
- ⑪ **只要模块之间的接口关系不变，各模块的内部实现细节的修改不会影响别的模块**。
这是**正确**的。
- ⑫ 在编写程序时尽可能对程序代码进行优化。
这是**错误**的，**提前优化是不可取的**。
- ⑬ 在进行需求分析时，需同时考虑维护问题。
正确。
- ⑭ **编码时应尽可能使用全局变量**。
这是**错误**的，这将**使代码难以拓展和维护**。
- ⑮ **不要修补不好的程序，要重新写**。
正确。
- ⑯ 程序中的注释越少越好。
这是**错误**的，**关键代码位置必须有注释**，不然一周之后看自己的代码自己都觉得这是什么垃圾东西。
- ⑰ 程序的书写格式应有助于读者阅读与理解。
正确。
- ⑱ 软件的功能便于扩充。
正确。
- ⑲ 程序易理解，也便于排错。
正确。

6) 六部分

- ① 软件工程与计算机科学性质不同，软件工程着重于**建造软件系统**。
- ② 软件工程是一种**自顶向下**分阶段实现的软件开发方法。
- ③ 结构化方法和原型化方法在实际的应用中，它们之间的关系表现为**相互补充**。
- ④ 在软件生存周期的模型中，**螺旋模型**吸收了软件工程“演化”的概念。
- ⑤ 技术可行性要解决**技术风险问题**。
- ⑥ 软件需求分析阶段的工作，可以分成一下四个方面，**对问题的识别，分析和综合，指定规格说明以及需求分析评审**。
- ⑦ 好的软件结构应该是**低耦合，高内聚**。
- ⑧ 流程图中的顺序结构中各个方框是对程序的**逻辑意义**进行分块，使之表达的更清晰。

- ⑨ 软件可靠性表明一个程序按照用户的要求和设计目标，执行其功能的正确程度即“软件可靠性”是在给定的时间间隔及给定的设计要求下，成功地运行程序的概率。
- ⑩ 集成测试时，能较早发现高层模块接口错误的测试方法为自顶向下渐增式测试。
- ⑪ 软件概要设计包括软件系统结构设计以及数据结构和数据库设计。
正确。
- ⑫ 设计质量的评审对象是在详细设计阶段产生的软件需求规格说明、数据要求规格说明，在软件概要设计阶段产生的软件概要设计说明等。
错误，应该是需求分析阶段。
- ⑬ 在软件生存期每个阶段的工作中都可能引入人为的错误。当出现错误，即使不及时纠正，也不会传播到开发的后续阶段中去，并在后续阶段中引出更多的错误。错误，有错误就应该及时处理。
- ⑭ 软件测试的目的是尽可能多的找出软件的缺陷。
正确。
- ⑮ 模块化，信息隐藏，抽象和逐步求精的软件设计原则有助于得到高内聚，低耦合度的软件产品。
正确。
- ⑯ 单元测试又称为白盒测试，单元测试是对软件设计的最小单位——程序模块进行正确性检验的测试。
错误，单元测试不是白盒测试。
- ⑰ 软件需求的逻辑视图给出软件要达到的功能和要处理数据之间的关系，也是实现的细节。
错误，并不是实现的细节。
- ⑱ 确认测试计划应该在可行性研究阶段制定。
错误，是验收阶段。
- ⑲ 软件产品的质量是软件工程的开发工作的关键问题，也是软件工程生产中的核心问题。
正确。
- ⑳ 重视程序结构的设计，使程序具有较好的层次结构。
正确。

2. 长句整理

1) 第一部分

调用算法是解决问题的关键步骤之一，算法的计算工作量大小和实现算法所需的存储单元多少，分别称为计算的 α 复杂度和 β 复杂度，编写程序时，程序的结构化和使用有实际意义的名字是应采纳的原则之一。排错是调试程序的主要工作之一。

2) 第二部分

软件测试通常分为，单元测试，集成测试，确认测试和系统测试，其中确认测试主要用于发现需求分析阶段的错误。在集成测试时，通常可采用自顶向下渐增式集成，在自底向上渐增式集成时，对每个被集成的模块要设计驱动模块，但不必设计桩模块。通常还要进行 α 测试和 β 测试，以发现可能只有最终用户才能发现的错误， α 测试是指最终用户在开发环境下，开发人员在场所进行的测试， β 测试是指最终用户在实际使用环境下，开发人员不在场所进行的测试。在软件维护阶段，当修改软件后，除了进行常规的测试外，还应进行回归测试，

3) 第三部分

黑盒测试法是通过分析程序的**功能**来设计测试用例的方法，除了测试程序外，它还适用于对**需求分析**阶段的软件文档进行测试。白盒测试法是根据程序的**内部逻辑**来设计测试用例的方法，除了测试程序外，他也适用于对**软件详细设计**阶段的软件文档进行测试。白盒测试法测试时按照给定的覆盖条件选取测试用例，**组合条件**覆盖比其他条件都要严格，但它不能保证覆盖程序中的每一条路径。

4) 第四部分

软件过程不断改进是**软件工程七原理**基本原理之一，是**软件生存期**的基本过程之一，软件过程需要不断完善，首先从**非工程化**的软件开发方式改变为**工程化**的软件开发方式。按照软件过程的系统方法进行软件的工程活动和管理活动，进而不断完善软件各个软件过程，从而不断提高**软件过程能力**，软件过程能力的提高首先需要对当前的软件过程状况进行科学的评估。

5) 对五部分

程序的三种基本控制结构是**顺序**，**条件**，**循环**，他们共同点是**只有一个入口和一个出口**，结构化程序设计的一种基本方法是**逐步求精法**。软件测试的目的是**发现程序中的错误**。软件测试得目的是**找出错误所在并改正之**。

3. 概念陈述

1) 父图与子图平衡

数据流图中要求父图的输入数据流、输出数据流与子图的输入数据流、输出数据流完全相同。

2) 模块化

是指解决一个复杂问题时自顶向下逐层把软件系统划分成若干模块的过程。

3) 脚本

是系统在一个特定执行期间所发生的事件序列。

4) 动态模型

动态模型描述与操作时间和顺序有关的系统特征、影响更改的事件、事件的序列、事件的环境以及事件的组织。

5) 可测试性

是指软件发现故障并隔离、定位其故障的能力特性，以及在一定的时间和成本前提下，进行测试设计、测试执行的能力。

6) 软件

是计算机系统中与硬件相互依存的另一部分，它是包括程序、数据、相关文档的完整集合以及后续的维护服务。

7) 白盒测试

是对软件的过程性描述做细致的检查，软件测试员利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序所有逻辑路径进行测试。

8) 边界值分析

就是首先确定边界情况，然后选取正好等于、刚刚大于，或刚刚小于边界的值作为测试数据。

9) 模块的扇入/扇出数

在软件结构中，下级模块被上级模块调用的个数称为模块的扇入，上级调用下级模块的

个数称为扇出数。

10) 静态模型

是通过系统对象类及其之间的关系描述系统的静态结构。

11) 确认

我们是否在建成一个正确的产品。(强调结果正确)

12) 验证

我们是否在正确地建造一个产品。(强调过程正确)

13) 增量模型

该模型是一种非整体开发的模型。软件在该模型中是“逐渐”开发出来的，开发出一部分，向用户展示一部分，可让用户及早看到部分软件，及早发现问题。

14) 继承

就是子类自动共享父类数据结构和方法的机制。

15) 多态

在收到消息时对象要予以响应，不同的对象收到同一消息可以产生完全不同的结果。

16) 等价类

指某个输入域的子集合。在该子集合中，各个输入数据对于揭露程序中的错误都是等效的。

17) 功能内聚

是指模块内部各个成分之间的关系，又称为块内联系，是模块功能相对强度的度量。

18) α 测试

是由一个用户在开发环境下进行的测试，也可以是开发机构内部的用户在模拟实际操作环境下进行的测试。

19) β 测试

是由软件的多个用户在一个或多个用户的实际使用环境下进行的测试。

20) 测试

是为了发现错误。

21) 调试

是要找出错误的位置并改正错误。

22) 程序的可维护性

为满足用户新的需求，或当环境发生了变化，或运行中发现了新的错误时，对一个已投入运行的软件进行相应诊断和修改所需工作量的大小。

23) 模块的控制范围

指该模块及所包含的子树。

24) 容错技术

对那些无法避开的差错，使其影响减少至最小的技术。也就是说，当错误发生时，尽可能地不影响其它的系统元素，或是把用户的影响限制在某些容许的范围内。

25) 驱动模块与桩模块

在测试过程中，用以代替被测试模块的上级模块称为**驱动模块**，用以替代被测试模块的下级模块称为**桩模块**。

26) 渐增式测试

在对系统的测试过程中,将模块是按照测试方案,一个一个逐步添加到被测试的系统中。

27) 信息隐藏

是指在设计和确定模块时,使得一个模块内包含的信息(过程或数据)对于不需要这些信息的模块来说是不能访问的。

28) 软件工程过程

为获得软件产品,在软件工具的支持下由软件工程师完成的一系列软件工程活动。

4. 简答

1) 快速原型技术的基本思想是什么

在软件开发的早期,快速开发一个目标软件系统的原型,让用户对其进行评价并提出修改意见,然后开发人员根据用户的意见对原型进行改进。

2) 为什么要进行需求分析?通常对软件系统有哪些需求?

需求问题是软件工程项目失败的主要原因。一个高质量的软件,很大程度上取决于对要解决的问题的认识,以及如何准确的表达用户的需求。需求分析就是要解决这个问题。通常,对一个软件系统的需求可以有功能需求、性能需求、可靠性和可用性需求、出错处理需求、接口需求等多个方面。

3) 为什么说文档是影响软件可维护性的决定因素

由于长期使用的软件系统在使用过程中会有多次修改的经历,而文档应该记载了这些重要的修改,所以文档比之代码更为重要。

4) 什么是序言性注释,它包涵哪些内容

序言性注释通常放在每个程序模块的开头部分,它应当给出程序的整体说明,对于理解程序具有引导作用。序言性注释一般包含下列内容。

- ① 程序标题:有关该模块的功能和目的说明;主要算法;
- ② 接口说明:包括调用形式,参数描述、子程序清单;
- ③ 有关数据描述(重要的变量及其用途,约束或限制条件,以及其他有关信息);
- ④ 模块位置(在哪一个源文件中,或隶属于哪一个软件包);
- ⑤ 开发简历 模块设计者,复审者,复审日期,修改日期及有关说明等。

5) 功能模型的特征有哪些

功能模型**描述了系统的所有计算**。功能模型指出发生了什么,对动态模型确定什么时候发生,而对象模型确定发生的客体。功能模型表明一个计算如何从输入值得到输出值,它不考虑所计算的次序。功能模型由多张数据流图组成。数据流图说明数据流是如何从外部输入、经过操作和内部存储输出到外部的。功能模型也包括对象模型中值的约束条件。

6) 什么是测试用例以及动态测试有哪些方法

所谓“测试用例”是指**为寻找程序中的错误而精心设计的一组测试数据**,每个测试用例一般是一个二元组(输入数据、预期结果)。动态测试主要有两种方法:白盒测试法和黑盒测试法。

白盒测试法是从被测试程序的内部逻辑入手来设计测试用例;**黑盒测试法**着重测试被测试程序的功能,而**不关心内部如何实现其功能的结构**,是从用户观点出发的测试。但无论白盒测试法还是黑盒测试法都只能选择一些有代表性的测试用例进行有限的测试。

a. 白盒测试法

有 6 种用于有判定存在的逻辑覆盖标准，即程序内部路径的覆盖程度。对于有循环存在的覆盖用限制循环次数的办法来测试。但是对于一个实际问题的程序测试中，其路径是一个庞大的数字，基本路径测试告诉了最少路径的计算办法，在分析程序控制流程图环路复杂性的基础上，导出基本路径集合，从而设计测试用例，保证这些基本路径至少执行一次，以最少的用例发现尽量多的错误。

b. 黑盒测试法

具体方法有 4 种，采用黑盒测试法的综合策略是先用等价类划分法（包括边界值分析法，即取边值上的数）设计出测试用例，然后用错误推测法补充。如果被测试程序含有多个条件的逻辑组合，则开始就用因果图法。

7) 怎样克服软件危机

充分吸收和借鉴人类长期以来从事各种工程项目中积累的行之有效的有效原理、概念、技术与方法，特别是吸取几十年来人类从事计算机硬件研究和开发的经验教训。在开发软件的过程中努力作到良好的组织，严格的管理，相互友好的协作。

推广在实践中总结出来的开发软件的成功的技术和方法，并研究更好、更有效的技术和方法，尽快克服在计算机系统早期发展阶段形成的一些错误概念和作法。

根据不同的应用领域，开发更好的软件工具并使用这些工具。将软件开发各个阶段使用的软件工具集成成一个整体，形成一个很好的软件开发支撑环境。

总之为了解决软件危机，既要有技术措施(方法和工具)，又要有必要的组织管理措施。

8) 说明软件工程管理的重要性

由软件危机引出软件工程，是计算机发展史上的一个重大进展。为了对付大型复杂的软件系统，须采用传统的“分解”方法。**软件工程的分解是从横向和纵向即空间和时间两个方面进行的**。横向分解就是把一个大系统分解为若干个小系统，小系统分解为子系统，子系统分解为模块，模块分解为过程。纵向分解就是生存期，把软件开发分为几个阶段，每个阶段有不同的任务、特点和方法。为此，软件工程管理需要有相应的管理策略。根据软件产品的特征，且随着软件规模的不断增大，开发人员也随之增多，开发时间也相应地持续增长，这些都增加了软件工程管理的难度，同时也突出了软件工程管理的必要性与重要性。事实证明，由管理失误造成的后果比程序错误造成的后果更为严重。很少有软件项目的实施进程能准确地符合预定目标、进度和预算的，这也就是以说明软件工程管理的重要性。

9) 请说明多态重用与继承重用的关系

多态重用实际上是一种特殊的继承重用，是充分利用多态性机制支持的继承重用。一般来说，使用多态重用方式重用已有的类构件时，在子类中需要重新定义的操作比较少，因此，**多态重用方式比继承重用方式成本低**。

10) 用户界面设计的最一般原则，即所谓“黄金规则”是什么

置用户于控制之下 - 减少用户的记忆负担 - 保持界面一致

11) 画出软件生存周期的瀑布模型并简述各部分的主要任务

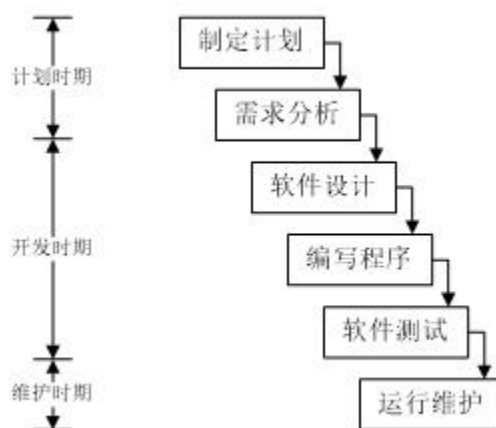


图 1. 瀑布模型

a. 制定计划

由系统分析员根据对问题的理解，提出关于《系统目标与范围的说明》，请用户审查和认可。

b. 需求分析

弄清用户对软件系统的全部需求，并用《需求规格说明书》的形式准确地表达出来。

c. 软件设计

建立软件的总体结构，画出由模块组成的《软件结构图》或《层次图》。根据总体结构进行单个模块的设计，确定模块内部的过程结构。

d. 编写程序

按照选定的语言，把模块的过程性描述翻译为源程序。

e. 软件测试

是保证软件质量的重要手段，可尽量地减少软件中的隐藏错误。

f. 运行维护

是使软件在整个生存周期内保证满足用户的需求和延长软件的使用寿命。

12) 白盒测试法有哪些覆盖标准，试对它们的检错能力进行比较

白盒测试法的覆盖标准有：语句覆盖、判定覆盖、条件覆盖、条件组合覆盖和路径覆盖。

前五种覆盖标准把注意力集中在单个判定或判定的各个条件上,可能会使程序某些路径没有执行到。路径覆盖根据各判定表达式取值的组合,使程序沿着不同的路径执行,查错能力强。但由于它是从各判定的整体组合出发设计测试用例的,可能使测试用例达不到条件组合覆盖的要求。

13) 比较程序流程图与 PAD 图的特点

这两种图均为详细设计阶段使用的描述工具。

a. 程序流程图

直观清晰,易于使用,但控制流程线不易限制,不易反映逐步求精的过程,不易表示数据结构。

b. PAD 图

是一种由左往右展开的二维树型结构,其能清晰地反映程序的层次结构,支持逐步求精的设计方法,易读易写,使用方便;可自动生成程序,支持结构化程序设计原理。

14) 可行性研究报告主要包括哪些内容

可行性分析的内容一般包含:项目的背景、系统概要、候选方案、技术上的可行性分析、经济上可行性分析、法律上可行性分析、操作上的可行性分析。

15) 可行性研究报告的结论一般有哪几种

可行性分析的结论一般有,可行或不可行两种。

16) 什么是功能内聚,什么是内容耦合

a. 功能内聚

是指模块内包括并仅包括为完成某个功能所必须的所有成分。或者说模块内的所有成分共同完成一个功能,缺一不可。

b. 内容耦合

是指一个模块直接访问另一模块的内部信息(程序代码或内部数据)