

软件工程常考简答题

绪论

1、什么是软件工程？软件工程和计算机科学有何区别？

答：**软件工程**是以计算机科学理论及其他相关学科的理论为指导，采用工程化的概念、原理、技术和方法进行软件的开发和维护，把经过时间证明是正确的管理措施和当前能够得到的最好的技术、方法相结合，以期用较少的代价获取高质量的软件。

计算机科学侧重于理论研究，其成果可应用于软件工程，而软件工程则强调如何有效的建造一个软件系统。

2、什么是软件？

答：**计算机软件**是与计算机系统操作有关的**程序、规程、规则**及任何与之有关的**文档及数据**；

3、什么是软件危机？分析其产生的原因与摆脱危机的方法，你认为现在已经摆脱软件危机了吗？请说明理由。

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。主要是两个问题：（1）如何开发软件，怎样满足对软件的日益增长的需求；（2）如何维护数量不断膨胀的已有软件；

软件危机的出现，究其原因，一方面是由软件本身的特点引起的，如软件越来越复杂，软件故障难于检测，工作量难以估计；另一方面则是因为软件开发和维护方法不当所造成的，如软件的个体化特征太强等。

为了摆脱软件危机所造成的困境，提出软件工程的**概念**，其主要思路是：要把人类长期以来从事各种工程项目所积累起来的行之有效的原理，概念，技术和方法，特别是人类从事计算机硬件研究和开发的经验教训，应用到软件的开发和维护中来。

我认为我国现在还未完全摆脱软件危机，因为随着经济发展，各行各业对软件人才的需求不断扩大，由于我国的软件人才缺口依旧很大，造成了某些行业内软件开发进度难以预测，软件开发成本难以控制，用户对产品功能难以满足，软件产品质量无法保证，软件产品难以维护，软件缺少适当的文档资料。

4、软件产品的特性是什么？

软件是一种逻辑产品，具有**无形性**；软件产品的生产主要是**研制**；软件不存在磨损和老化问题，但存在退化问题；软件产品的生产主要是**脑力劳动**；软件产品生产成本昂贵，目前生产方式还未摆脱手工生产方式；软件具有“**复杂性**”，其开发和运行常受到计算机系统的限制。

5、软件发展有几个阶段？各有何特征？

①程序设计阶段

硬件特征：价格贵、存储容量小、运行可靠性差。

软件特征：只有程序、程序设计概念，不重视程序设计方法。

②程序系统阶段。

硬件特征：速度、容量及工作可靠性有明显提高，价格降低，销售有爆炸性增长。

软件特征：程序员数量猛增，开发人员素质低。

③软件工程阶段。

硬件特征：向超高速、大容量、微型化及网络化方向发展。

软件特征：开发技术有很大进步，但未获得突破性进展，软件价格不断上升，未完全摆脱软件危机。

6、试简要叙述文档在软件工程中的作用。

在软件工程中，文档的作用是：

- (1) 提高软件开发过程的能见度；
- (2) 记录开发过程的有关信息便于使用与维护；
- (3) 作为开发人员阶段工作成果和结束标志；
- (4) 提高开发效率；
- (5) 提供软件运行、维护和培训有关资料；
- (6) 便于用户了解软件功能、性能。

7、简述一个成熟的软件机构应具有的特点

建立了机构级的软件开发与维护过程。软件人员对其有较好的理解。一切活动均遵循过程的要求进行，做到工作步骤有次序，且有章可循；软件过程必要时可做改进，但需在经过小型实验和成本效益分析的基础上进行；软件产品的质量和客户对软件产品的满意程度不是由开发人员，而是由负责质量保证的经理负责监控；项目进度和预算是根据以往项目取得的实践经验确定，因而比较符合实际情况。

8、软件生存周期中的软件开发过程由哪几个阶段组成？各阶段的任务是什么？

答：以瀑布模型为例主要包括六个阶段：

- ①可行性研究，确定系统的目标与规模，分析项目的可行性
- ②需求分析与规格说明，明确系统的规格与要求
- ③设计，包括概要设计与详细设计
- ④编程，用程序语言实现每个模块的功能
- ⑤测试，发现并改正错误
- ⑥运行与维护

9、瀑布模型相关概念。

瀑布模型的基本概念以及特点。瀑布模型的缺陷有哪些？

瀑布模型规定了各项软件工程活动，包括需求分析、规格说明、设计、编码、测试和维护，并规定了它们自上而下、相互衔接的固定次序，如同瀑布流水，逐级而下。

瀑布模型的特点是：阶段间具有顺序性和依赖性；清楚区分逻辑设计和物理设计，尽可能推迟程序的物理实现；每个阶段都必须完成规定的文档，且每阶段结束前需要对完成的文档进行评审。

缺陷：(1)完全地、精确地刻画复杂应用问题的用户需求是不可能的。(2)需求分析和软件实现是不可分割的。(3)用户和开发人员都难于理解形式的、不可执行的需求规格说明。(4)传统的开发策略没有为用户参与关键的开发决策提供便利。(5)目标软件的适应能力不强(6)瀑布模型并不适合于软件项目管理

10、螺旋模型相关概念。

(1) 简述螺旋模型的基本开发过程。螺旋模型哪些优点？

螺旋模型的基本思想是使用原型及其他方法来尽量降低风险。可以理解为在快速原

型模型的每个阶段之前都增加了风险分析。

基本开发过程：(1)需求定义。利用前面介绍的需求分析技术理解应用领域，获取初步的用户需求，制定项目开发计划。(2)风险分析。根据初始需求或改进意见评审可选用的方案，给出消除或减少风险的途径。(3)工程实现。利用前一节介绍的快速原型构造方法针对已知的用户需求生成快速原型。(4)评审。将原型提交用户使用并征询用户改进意见。上述过程将不断迭代，直至给出用户满意的目标软件产品。

优点：(1)螺旋模型支持用户需求的动态变化。(2)螺旋模型中，需求分析和软件实现是互相依赖、紧密联系的软件开发活动。(3)原型作为形式的可执行的需求规格说明，既易于为用户和开发人员共同理解，又可作为后续开发的基础。(4)为用户参与软件开发的所有关键决策提供了方便。(5)有助于提高目标软件的适应能力(6)为项目管理人员及时调整管理决策提供了便利，从而降低了软件开发风险。

(2) 为何螺旋模型主要适合开发人员和用户为同一个组织机构成员的软件项目的开发？

因为如果开发人员和用户属于不同的组织机构，则一般在正式的软件开发之前均需要签订合同，所有的风险分析均必须在签订合同之前完成。但螺旋模型需要在软件开发的每一个阶段都进行风险分析，当重要风险不能解决时，项目应该终止。由此螺旋模型可能因为风险使得合同在完成之前就终止，从而导致法律诉讼。故螺旋模型不适合开发人员和用户属于不同组织机构的情况。

11、快速原型模型相关概念。

(1) 快速原型技术的基本思想是什么？快速原型模型有几种？各有何特点？

基本思想：花费少量代价建立一个可运行的系统，使用户及早获得学习的机会，最终提高软件质量。

根据原型的不同作用，有**三类原型模型**：

(1)**探索型原型**。这种类型的原型模型是把原型用于开发的需求分析阶段，目的是要弄清用户的需求，确定所期望的特性，并探索各种方案的可行性。它主要针对开发目标模糊，用户与开发着对项目都缺乏经验的情况，通过对原型的开发来明确用户的需求。

(2)**实验型原型**。这种原型主要用于设计阶段，考核实现方案是否合适，能否实现，对于一个大型系统，若对设计方案心中没有把握时，可通过这种原型来证实设计方案的正确性。

(3)**演化型原型**。这种原型主要用于及早向用户提交一个原型系统，该原型系统或者包含系统的框或者包含系统的主要功能。在得到用户的认可后，将原型系统不断扩充演变为最终的软件系统，它将原型的思路扩展到软件开发的全过程。

(2) 快速开发原型的途径有哪些？

废弃型：快速建立需求规格原型；

追加型：快速建立渐进原型法采用循环渐进；

步骤：快速分析、构造原型、运行和评价原型、修改与改进

12、比较常用软件工程过程模型（瀑布模型，快速原型化模型，快速应用开发模型，渐增模型，螺旋模型）的优缺点。指出各自适用的情况。

瀑布模型：**优点：**包括强制性的分阶段的方法，要求每一阶段都必须提供相应的文档，因此开发过程便于管理和控制。瀑布模型是一种文档驱动模型，开发过程中产生的需求规格说明书文档，设计文档，代码文档，以及其他文档如数据库手册，用户手册和操作文档等，对于软件使用和维护是非常必要的。**缺点：**该模型将本来非线性的软件开发过程，人为地加以线性化，开发的产品只有交付使用时才能和用户见面，用户的反

反馈意见只能从需求分析开始从头修改，加大了软件开发的工作量。

快速原型化模型：**优点：**能够开发出真正满足用户需要的软件。**缺点：**若采用原型作为最终产品，不利于软件维护工作。

快速应用开发模型：**优点：**能够快速地完成整个信息系统的开发。**缺点：**对于大的项目，为建立适当数目的 RAD 开发小组可能需要大量的人力资源；同时，RAD 要求开发人员和用户双方在短期内对完成整个系统的开发达成协议，如果任何一方做不到的话，使用 RAD 进行开发则不可避免地会失败。

渐增模型：**优点：**每一轮都可以向用户分发布一个高质量的可操作的版本，用户不需要等到最终软件产品被开发出来就可以使用软件的部分功能。由于软件产品不是突然一次性提交给用户使用的，用户可以比较容易接受，并且容易提出中肯的意见。这种方式不要求非常大的初始资金投入。**缺点：**要求下一轮新增的功能应能够无缝地集成到下一轮系统中去，可能这一点并不容易做到。如果整体结构设计不当，可能会导致整个软件的结构变坏。

螺旋模型：**优点：**它强调可替换性和强约束性，使得螺旋模型支持对现有软件的复用，并且支持始终把软件质量看作一个目标。开发和维护被视为同样的方式。

缺点：主要适合于内部开发。只适合于规模比较大的系统的开发。它是风险驱动的，只有当项目组的成员擅长于进行风险分析时，才有理由采用螺旋模型。

可行性研究

1、可行性研究报告的主要内容有哪些？

一个可行性研究报告的主要内容如下：

(1) **引言：**说明编写本文档的目的；项目的名称、背景；本文档用到的专门术语和参考资料。

(2) **可行性研究前提：**说明开发项目的功能、性能和基本要求；达到的目标；各种限制条件；可行性研究方法和决定可行性的主要因素。

(3) **对现有系统的分析：**说明现有系统的处理流程和数据流程；工作负荷；各项费用支出；所需要各类专业技术人员的数量；所需要各种设备；现有系统存在什么问题。

(4) **所建议系统的技术可行性分析：**所建议系统的简要说明；处理流程和数据流程；与现有的系统比较的优越性；采用所建议系统对用户的影响；对各种设备、现有软件、开发环境、运行环境的影响；对经费支出的影响；对技术可行性的评价。

(5) **所建议系统的经济可行性分析：**说明所建议系统的各种支出，各种效益；收益投资比；投资回收周期。

(6) **社会因素可行性分析：**说明法律因素，对合同责任、侵犯专利权、侵犯版权等问题的分析；说明用户使用可行性，是否满足用户行政管理、工作制度、人员素质的要求。

(7) **其他可供选择方案：**逐一说明其他可供选择的方案，并说明未被推荐的理由。

(8) **结论意见：**说明项目是否能开发；还需要什么条件才能开发；对项目目标有什么变动等。

2、可行性研究的概念？目标？基本内容？步骤？

3、成本分析方法？

4、系统流程图与数据流图的概念与区别？

需求分析

1、需求分析的概念？需求收集的方式？

2、数据流建模的相关概念。

3、如何画分层数据流图

总的原则是：至顶而下，逐层分解（画分层数据流图）：比较复杂的系统不能画在一张纸上，逐层分解的画法可以控制每一层的复杂度。**顶层**：将整个系统作为一个加工，描述系统边界（输入与输出。）**中间层**：表示某个加工分解为一组子加工，其中的子加工还需进一步分解。**底层**：由不再进行分解的基本加工组成。

4、常用的软件需求分析的方法有哪些？

结构化分析方法（SA）、面向对象分析方法（OOA）、功能分解法、信息建模法

5、简述需求分析工作可以分成哪四个方面？软件需求分析有哪三个基本原则？

需求分析阶段分成**四个方面**：**对问题的识别、分析与综合、制定规格说明和评审**。**三个基本原则**：必须能够表达和理解问题的数据域和功能域；必须按自顶向下、逐步分解的方式对问题进行分解和不断细化；要给出系统的逻辑视图和物理视图。

6、试举例说明什么是功能性需求，什么是非功能性需求？为何非功能性需求往往比功能性需求还要重要？

功能性需求就是软件要实现的功能需求分析。

非功能性需求，是指软件产品为满足用户业务需求而必须具有且除功能需求以外的特性。软件产品的非功能性需求包括系统的性能、可靠性、可用性、可维护性、有效性和可移植性等。

重要性：将非功能性需求的各项特点举例展开即可。

7、在需求分析和设计阶段建立原型的好处有哪些？

在需求分析和设计阶段建立原型的好处有：

- （1）增进开发人员和用户对系统需求的一致理解，明确软件需求，尤其是功能含糊的需求。
- （2）原型提供了一种有力的学习手段，能够使用户尽早接触系统。
- （3）原型可以用来识别或化解风险。
- （4）原型有的可以原封不动地成为产品，有的略加修改就可以成为最终系统的一个组成部分，这样有利于建成最终系统。

8、IDEF0 功能建模相关概念。

9、IDEF0 方法有什么特点？

IDEF0 方法的特点是：

- （1）采用方框和箭头等简单的图形符号描述系统的活动和数据流，描述活动所受到的约束条件及实现机制，清楚地反映了系统的功能。
- （2）采用严格的自顶向下、逐层分解的方式建立系统功能模型。顶层确定系统范围，采用抽象原则，然后有控制的逐步展开有关活动的细节，符合 SA 方法的分析策略。同时，IDEF0 规定每张图至少有 3 个、最多有 6 个方框，上界 6 保证采用层次性描述复杂问题的可理解性，下界 3 保证分解有意义。

9、IDEF1X 建模相关概念。

11、需求规格说明的概念。

概要设计

1、概要设计中相关的概念。

2、软件设计阶段的任务与目标是什么？

任务：将分析阶段获得的需求说明转换为计算机中可实现的系统；即：软件体系结构的设计、用户界面的设计、数据结构的设计、算法的设计。

目标：构造一个高内聚低耦合的软件模型。

3、软件设计的准则是什么？

(1)、**软件结构的准则：**顶部宽度小，中部宽度最大，底部宽度次之；在结构顶部有较高的扇出数，在底部有较高的扇入数。

(2)、**模块化准则：**将系统划分成若干模块，每个模块完成一个子功能，把这些模块集中起来组成一个整体，以满足所要求的整个系统的功能。

(3)、**软件独立性准则：**开发具有功能专一，模块之间无过多相互作用的模块。

4、模块内联系与模块间联系有哪些种类？

模块间的联系分类法：按耦合度

如：内容耦合：一个模块直接引用另一模块的内部信息

模块内的联系方法：按内聚度依次展开

5、照“降低块间联系，提高块内联系”的设计总则对模块进行修改，具体从哪些方面进行改进？

答：1) 尽可能建立功能模块；2) 消除重复功能；3) 模块的作用范围与控制范围，即当作用范围为控制范围的子集时，才能获得较低的块间联系；4) 模块的大小适当；5) 模块的扇入扇出数不宜太多。

6、衡量模块独立性的两个定性标准是什么（2分）？这两个标准的定义分别是什么（2分）？在我们的软件设计中，关于模块独立性我们追求的目标是什么（1分）？

衡量模块独立性的两个**定性标准**是内聚和耦合（2分）。**耦合**是指对一个软件结构内不同模块彼此之间互相依赖（连接）的紧密程度；而**内聚**则标志一个模块内部各个元素彼此结合的紧密程度。（2分）。在我们的软件设计中，关于模块独立性我们追求的目标是**紧密内聚松散耦合**（1分）。

详细设计

1、详细设计的目标与任务。

2、详细设计的图形描述工具，及各自的概念和优缺点。

3、Jackson 方法的概念。

4、程序规格说明。

5、层次图（即软件结构图）和层次方框图（即 Jackson 图）有何区别？

相同：形式类似。

不同：1,层次图描绘软件的层次结构.层次方框图描绘的是数据结构。

2,层次图的方框表示模块或子模块。层次方框图的方框表示数据结构整体或其子集。

3,层次图的连线表示调用关系，层次方框图表示组成关系。

层次图主要用来描绘软件的层次结构，矩形框代表一个模块，矩形框之间的连线表示模块之间的调用关系，常与 IPO 图一起形成 HIPO 图。层次方框图主要描述系统的组成关系，一般自用域需求分析。

软件测试

1、软件测试的概念、目的。

2、软件测试对象应包括哪些内容？

软件测试并不等于程序代码测试，其应该贯穿于软件定义与开发的整个期间。因此，需求分析、概要设计以及程序编码等各阶段所得的文档，包括需求规格说明、概要设计说明、详细设计规格说明以及源程序代码都应成为软件测试的对象。

3、白盒测试的概念及相关技术。

4、黑盒测试的概念及相关技术。

5、等价分类法相关概念。

(1) 等价分类法的基本思想是什么？

根据程序的输入特性，将程序的定义域划分为有限个等价区段——“等价类”，从等价类中选择出的用例具有“代表性”，即测试某个等价类的代表值就等价于对这一类其它值的测试。如果某个等价类的一个输入数据（代表值）测试中查出了错误，也即是该等价类中的其它值也会查出同样的错误。

(2) 等价类划分方法中，划分好等价类后设计测试用例的步骤是什么？

1) 为每个等价类编号；2) 使一个测试用例尽可能覆盖多个有效等价类；

3) 特别要注意：一个测试用例只能覆盖一个无效等价类；

6、软件测试为何需要尽早和不断地进行？

软件测试不应仅仅作为一个独立的软件开发阶段，而应贯穿到软件开发的各个阶段中。通过各阶段的评审，在早期把错误解决，以减少错误放大效应，这样不仅可以提高软件质量，而且也是降低软件成本的一个重要措施。由此，软件测试应该尽早和不断地进行。

7、一般，驱动模块比桩模块容易设计，为什么？

因为**驱动模块**是模拟主程序或者调用模块的功能，处于被测试模块的上层，所以驱动模块只需要模拟向被测模块传递数据，接收、打印从被测模块返回的数据的功能，较容易实现。而**桩模块**用于模拟那些由被测模块所调用的下属模块的功能，由于下属模块往往不只一个，也不只一层，由于模块接口的复杂性，桩模块很难模拟各下层模块之间的调用关系，同时为了模拟下层模块的不同功能，需要编写多个桩模块，而这些桩模块所模拟的功能是否正确，也很难进行验证。因此，驱动模块的设计显然比桩模块容易。

8、什么是 α 测试和 β 测试？

α 测试是在开发机构的监督下，由个别用户在确认测试阶段后期对软件进行测试，目的是评价软件的 FLURPS（功能、局域化、可使用性、可靠性、性能和支持性），注重界面和特色。 **β 测试**是由支持软件预发行的客户对 FLURPS 进行测试，主要目的是测试系统的可支持性。

9、渐增式与非渐增式各有何优、缺点？为什么通常采用渐增式？

非渐增式是将所有的模块一次连接起来，简单、易行、节省机时，但测试过程难于排错，发现错误也很难定位，测试效率低；**渐增式**是将模块一个一个的连入系统，每连入一个模块，都要对新系统进行测试。这种组装测试相比于非渐增式容易查出错误，并及时定位，有利于查出模块接口部分的错误，因此测试效率高。故通常采用渐增式测试方法。

10、什么是动态测试？为何静态测试方法往往比动态测试方法效率高？

动态测试是通过运行软件来检验其动态行为和运行结果的正确性。

动态测试是实际运行被测程序，输入相应的测试用例，判定执行结果是否符合要求，从

而检验程序的正确性、可靠性和有效性。

静态测试是对被测程序进行特性分析的一些方法的总称，这种方法的主要特性是不利用计算机运行被测试的程序，而是采用其他手段达到检测的目的。**因而在代码产生之前便可对各种文档进行测试，其效率当然高于动态测试。**

11、单元测试与集成测试相关概念。

(1) 请简述单元测试的主要内容？

模块接口测试、局部数据结构测试、路径测试、错误处理测试、边界测试

(2) 在单元测试中，所谓单元是如何划分的？

单元测试的对象通常是软件设计的最小逻辑单元，单元的划分在面向过程的结构化程序中一般是函数或子过程，在面向对象的程序中可以是类或类的成员函数。

(3) 如果开发时间紧迫，是否可以跳过单元测试而直接进行集成测试？

不可以。因为没有经过单元测试的模块会遗留大量的缺陷到集成测试阶段，而在集成测试阶段对这些缺陷定位困难，导致后续工作展开困难，修复缺陷成本成指数级增长。

(4) 集成测试也叫组装测试又叫联合测试，为何要进行集成测试？

在单元测试的基础上，将所有模块按照设计要求（如根据结构图）组装成为子系统或系统，进行集成测试。实践表明，一些模块虽然能够单独地工作，但并不能保证连接起来也能正常的工作。程序在某些局部反映不出来的问题，在全局上很可能暴露出来，影响功能的实现。

12、如果没有软件规格说明或需求文档，可以进行动态黑盒测试吗？为什么？

答：不行。因为黑盒测试是基于软件规格说明的测试。

13、软件测试阶段如何划分？

答：软件测试的阶段划分为：规格说明书审查；系统和程序设计审查；单元测试；集成测试；确认测试；系统测试；验收测试。

14、分析比较面向对象的软件测试与传统的软件测试的异同。

解：**传统的单元测试**的对象是软件设计的最小单位——模块。当考虑面向对象软件时，单元的概念发生了变化，此时最小的可测试单位是封装的类或对象，而不再是个体的模块。传统单元测试主要关注模块的算法实现和模块接口间数据的传递，而面向对象的单元测试主要考察封装在一个类中的方法和类的状态行为。

面向对象软件没有层次的控制结构，因此传统的自顶向下和自底向上集成策略就不再适合，它主要有以下两种集成策略：基于类间协作关系的横向测试；基于类间继承关系的纵向测试。系统测试一般不考虑内部结构和中间结果，因此面向对象软件系统测试与传统的系统测试差别不大。

面向对象软件测试的整体目标和传统软件测试的目标是一致的，即以最小的工作量发现尽可能多的错误，但是面向对象测试的策略和战术有很大不同。测试的视角扩大到包括复审分析和设计模型，此外，测试的焦点从过程构件(模块)移向了类。

软件维护

1、软件维护的概念、目的、维护类型。

2、软件维护衍生的一些相关技术。

3、为什么要进行软件维护？

软件维护是指软件系统交付使用以后，为了改正错误或满足新的需求而修改软件的过程。一个中等规模的软件，如果其开发过程需要一年时间，则它投入使用后，其运行的时间可能持续五到十年。在这个维护阶段，人们不仅需要着手解决开发阶段尚未解决的问题，同时，还需解决维护过程本身产生的问题。维护阶段的时间大约占整个软件生命周期的67%。因此，我们应充分认识到软件维护的重要性。

4、怎样防止维护的副作用？

修改代码所产生的副作用一般可以在回归测试过程进行查明和纠正；**修改软件所带来的副作用**可以通过完善的设计文档的加以限制；**修改文档资料所产生的副作用**可以在软件系统交付使用之前，对整个软件配置进行复审，将能大大减少。

5、提高软件可维护性的方法有哪些？

- (1) 建立明确的软件质量目标与优先级。
- (2) 使用提高软件质量的技术与工具（如模块化）
- (3) 进行明确的质量保证审查
- (4) 选择可维护性好的程序设计语言
- (5) 改进程序文档
- (6) 开发时考虑到维护

软件质量

1、软件质量保证应做好以下几方面的工作：(1) **采用技术手段和工具。**质量保证活动要贯彻开发过程始终，必须采用技术手段和工具，尤其是使用软件开发环境来进行软件开发。(2) **组织正式技术评审。**在软件开发的每一个阶段结束时，都要组织正式的技术评审。国家标准要求单位必须采用审查、文档评审、设计评审、审计和测试等具体手段来保证质量。(3) **加强软件测试。**软件测试是质量保证的重要手段，因为测试可发现软件中大多数潜在错误。(4) **推行软件工程规范（标准）。**用户可以自己制定软件工程规范（标准），但标准一旦确认就应贯彻执行。(5) **对软件的变更进行控制。**软件的修改和变更常常会引起潜伏的错误，因此必须严格控制软件的修改和变更。(6) **对软件质量进行度量。**即对软件质量进行跟踪，及时记录和报告软件质量情况。

2、什么是软件配置管理？软件配置管理和软件维护有何不同？

软件配置管理是在软件生命周期内管理变化的一组活动，用来：标识变化、控制变化、确保适当地实现了变化、向相关人员报告变化。

软件配置管理和软件维护的区别是：软件配置管理是一组追踪和控制活动，在软件项目启动时就开始，并一直持续到软件被淘汰后才终止；软件维护是一组软件工程活动，发生于软件交付给用户并投入运行之后。

面向对象方法和结构化方法

1、结构化程序设计的优点是什么？如何在编码中使用这种方法？

答：易于阅读，易于验证。程序设计自顶向下，逐步求精

2、SP 法的自顶向下，逐步求精方法的优点有哪些？

答：此法符合人们解决复杂问题的普遍规律，可提高软件开发的成功率和生产率；而且用先全局后局部、先整体后细节、先抽象后具体的逐步求精过程，开发出来的程序具有清晰的层次结构，因此程序容易阅读和理解，这样，程序自顶向下，逐步细化，分解成一个树型结构。因此，在同一层次的节点上做细化工作，相互之间没有联系，它们之间的细化工作相互独立；任何一步发生错误，一般只影响它下层的节点，同一层其他节点不受影响；在以后的测试中，也可以先独立地一个节点一个节点的做，最后再集成。这样，程序清晰和模块化，使得在修改和重新设计一个软件时，可复用的代码量最大。

3、简述结构化程序设计方法的基本特点。（5分）

- 1). 是尽可能少用goto语句的程序设计方法；
- 2). 每个代码块都单入单出；
- 3). 使用控制结构：顺序、选择和循环+Do_UNTIL和Do_CASE；
- 4). 自顶向下逐步求精。

4、面向对象方法学包含哪些核心概念？简述之

- (1) 对象：对象是现实世界中个体或事物的抽象表示，是其属性和相关操作的封装。
- (2) 类：类表示某些对象在属性和操作方面的共同特征。
- (3) 继承：类之间的继承关系是现实世界中遗传关系的模拟，它表示类之间的内在联系以及对属性和操作的共享。
- (4) 聚集：现实世界普遍存在部分—整体关系。
- (5) 消息：消息传递是对象与其外部世界相互关联的唯一途径。

5、面向对象程序设计有哪些优点？

答：并发时间短，效率高，可靠性高，所开发的程序更强壮。由于面向对象编程的可重用性，可以在应用程序中大量采用成熟的类库，从而缩短了开发时间。程序更易于维护、更新和升级。继承和封装使得应用程序的修改带来的影响更加局部化。

6、面向对象方法和结构化方法有何区别和联系？

答：**区别**：结构化方法是一种传统的开发方法，总的指导思想是自顶向下、逐步求精、单入口、单出口，基本原则是抽象和功能分解。从方法论上来看，结构化方法认为系统是由一些功能的相互联系、相互作用而形成的。

面向对象方法的出发点和基本原则是尽可能模拟人类习惯的思维方式，是开发软件的方法与过程尽可能接近人类认识世界的方法和过程。从方法论上来看，面向对象方法认为系统是由一些对象的相互联系、相互作用而形成的。

联系：面向对象开发是基于结构化开发的，还保留着结构化开发中的“函数”、“结构体”等元素。

8、试简要说明面向对象方法和结构化方法相比有何优越性，至少列出三点？

答：1、稳定性好； 2、可重用性好； 3、较易开发大型软件产品； 4、可维护性好。

UML

1、简述统一建模语言(UML)

统一建模语言(UML)是一种绘制软件蓝图的标准语言。可以用 UML 对软件密集型系统的制品进行可视化详述和文档化。UML 是一种定义良好、易于表达、功能强大且普遍适用的可视化建模语言（1 分）。它融入了软件工程领域的新思想、新方法和新技术。它的作用域不限于支持面向对象的分析与设计（1 分），还支持从需求分析开始的软件开发的全过程（1 分）。UML 的作用就是用很多图从静态和动态方面来全面描述我们将要开发的系统（2 分）。

2、UML 的消息类型有几种，哪几种？

4 种；（1）简单消息 （2）同步消息 （3）异步消息 （4）返回消息

3、UML 的特点是什么？

UML 的特点是：

- （1）统一了面向对象方法的有关概念和描述方法；
- （2）表达能力强，能对各种并发分布式系统进行建模，且 UML 还提供了扩展机制；
- （3）UML 专注于一种标准的建模语言，而不是一个开发过程。

4、什么是用例图？用例图有什么作用？

定义：由**参与者**（Actor）、**用例**（Use Case）以及它们之间的关系构成的用于描述系统功能的图成为用例图。（2 分）

作用：用例图是从软件需求分析到最终实现的第一步，它**显示了系统的用户和用户希望提供的功能**，有利于用户和软件开发人员之间的沟通（1 分）。用例图**可视化的表达了系统的需求**，具有直观、规范等优点，克服了纯文字性说明的不足（1 分）。用例方法是完全**从外部来定义系统的，它把需求 and 设计完全分离开来**（1 分），使用户不用关心系统内部是如何完成各种功能的。

5、试举例说明聚集关系中共享聚集和组合聚集有何不同。

如果在聚集关系中部分对象可同时参与多个整体对象的构成，则该聚集称为**共享聚集**。例如，一个课题组包含许多成员，每个成员又可以是另一个课题组的成员，则课题组和成员之间是一种共享聚集的关系。

如果部分对象完全隶属于整体对象，并与整体对象共存，则该聚集称为**组合聚集**。例如，在屏幕上一个窗口由文本框、列表框、按钮和菜单等构成，且一旦关闭了窗口，则各组成部分也消失，则窗口和它的组成部分之间的关系为组合聚集关系。