**第一份**

1、什么是软件工程？它目标和内容是什么？

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程目标：付出较低开发成本；达到要求的功能；取得较好的性能；开发的软件易于移植；只需较低的维护费用；能按时完成开发任务，及时交付使用；开发的软件可靠性高。

软件工程内容：包括开发技术和开发管理两个方面。

2、软件开发中有哪几种过程模型？哪些适用于面向对象的软件开发？

软件开发中常见的软件过程模型有瀑布模型、原型模型、螺旋模型、喷泉模型、统一软件过程等。

其中喷泉模型、统一软件过程适用于面向对象的软件开发。

3、什么是结构分析方法？该方法使用什么描述工具？

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。使用数据流图，数据字典，作为描述工具，使用结构化语言，判定表，判定树描述加工逻辑。

4、什么是软件概要设计？该阶段的基本任务是什么？

把一个软件需求转换为软件表示时，首先设计出软件总的体系结构。称为概要设计或结构设计。

基本任务：

⑴设计软件系统结构

⑵进行数据结构及数据库的设计

⑶编写概要设计的文档

⑷评审

5、软件维护有哪些内容？

（1）校正性维护。在软件交付使用后，一些隐含的错误在某些特定的使用环境下会暴露出来。为了识别和纠正错误，修改软件性能上的缺陷，应进行确定和修改错误的过程，这个过程就称为校正性维护。

（2）适应性维护。为了使应用软件适应计算机硬件、软件环境及数据环境的不断发生的变化而修改软件的过程称为适应性维护。

（3）完善性维护。为增加软件功能、增强软件性能、提高软件运行效率而进行的维护活动称为完善性维护。

（4）预防性维护。为了提高软件的可维护性和可靠性而对软件进行的修改称为预防性维护。

6、软件测试要经过哪些步骤?这些测试与软件开发各阶段之间有什么关系?

软件测试要经过的步骤是：单元测试→集成测试→确认测试→系统测试。

单元测试对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。该阶段涉及编码和详细设计文档。

集成测试是为了检查与设计相关的软件体系结构的有关问题，也就是检查概要设计是否合理有效。

确认测试主要是检查已实现的软件是否满足需求规格说明书中确定了的各种需求。

系统测试是把已确认的软件与其他系统元素(如硬件、其他支持软件、数据、人工等)结合在一起进行测试。以确定软件是否可以支付使用。

7、什么是软件生存周期？它有哪几个活动？

软件生存周期：一个软件从提出开发要求开始直到该软件报废为止的整个时期。

包括：可行性分析和项目开发计划，需求分析，概要设计，详细设计，编码，测试，维护。

8、衡量模块独立性的两个标准是什么？各表示什么含义？

内聚和耦合。

内聚：又称为块内联系，指模块内部各成分之间相互关联的程度，以高内聚为设计目标。

耦合：也称块间联系，模块之间相互联系程度的度量，联系越紧密，耦合性越强，独立性越差，以低耦合为设计目标。

9、什么是结构分析方法？结构化分析方法通过哪些步骤来实现？

结构化分析：简称SA，面向数据流进行数据分析的方法。采用自顶向下逐层分解的分析策略。顶层抽象地描述整个系统，底层具体地画出系统工程的每个细节。中间层则是从抽象到具体的过渡。

实现的步骤：⑴了解当前系统的工作流程，获得当前系统的物理模型。 ⑵抽象出当前系统的逻辑模型。 ⑶建立目标系统的逻辑模型。 ⑷作进一步补充和优化。

10、什么是需求分析？需求分析阶段的基本任务是什么？

需求分析：开发人员准确地理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的需求规格说明的过程。

基本任务：

⑴问题识别

⑵分析与综合，导出软件的逻辑模型

⑶编写文档

11、什么样是”事物流”？什么是”变换流”？

数据流图（DFD）一般可分为”变换流”，”事物流”两种。

̶变换流”：由输入﹑输出﹑变换（或称处理）三部分组成，是一顺序结构。

̶事物流”：它的某个加工，分离成许多发散的数据流，形成许多加工路径，并且根据输入值选择其中一个路径来执行。（这人加工称为事务处理中心）

12、单元测试有哪些内容?

单元测试主要针对模块的以下五个基本特征进行测试：

(1)模块接口

(2)局部数据结构

(3)重要的执行路径

(4)错误处理

(5)边界条件

13、什么是软件生存周期模型？有哪些主要模型？

生存周期模型：描述软件开发过程中各种活动如何执行的模型。对软件开发提供强有力的支持，为开发过程中的活动提供统一的政策保证，为参与开发的人员提供帮助和指导，是软件生存周期模型化技术的基础，也是建立软件开发环境的核心。

主要有：瀑布模型，增量模型，螺旋模型，喷泉模型，基于知识的模型，变换模型。

14、软件设计的基本原理包括哪些内容？

⑴模块化：模块是软件的组成部分，是具有独立功能且可命名的一段程序，所有模块组成整体，可以满足问题的要求。

⑵抽象：认识复杂现象过程所使用权的工具，只考虑事物本质的共同特性，忽略细节和其它因素。通过抽象确定组成软件的过程实体。

⑶信息隐蔽：将模块实现自身功能的细节与数据”隐蔽”起来。模块间仅交换为完成系统功能所必须的信息。

⑷模块独立性：每个模块只完成系统要求的独立的子功能。

15、结构化程序设计的基本要点是什么?

主要有三个：

1、采用自顶向下、逐步求精的程序设计方法

2、使用三种基本控制结构构造程序。任何程序都可以由顺序、选择、重复(循环)三种基本控制结构构造，这三种基本结构的共同点是单入口、单出口。

3、主程序员组的组织形式。

16、详细设计有哪几种描述方法？

详细设计的描述方法有图形、表格和语言，其中图形常用结构化程序流程图、盒图和PAD(问题分析图)为描述工具，语言常用过程设计语言(PDL)来作为工具。

17、单元测试中采用什么方法?

测试的方法是为被测试模块编写驱动模块和桩模块来实现被测试单元的可运行。通过驱动模块来模拟被测试模块的上级调用模块，以上级模块调用被测模块的格式驱动被测模块，接收被测模块的测试结构并输出。桩模块则用来代替被测试模块所调用的模块。它的作用是返回被测模块所需的信息。

18、什么是软件配置管理?什么是基线?

软件配置管理，简称SCM(Software Configuration Management)，是指一组管理整个软件生存期各阶段中变更的活动。软件配置管理技术可以使软件变更所产生的错误达到最小并最有效地提高生产率。

基线：是软件生存期中各开发阶段的一个特定点，它的作用是把开发各阶段工作的划分更加明确化，使本来连续的工作在这些点上断开，以便于检查与肯定阶段成果。

19、简述文档在软件工程中的作用？

文档在软件工程中的作用如下：

(1)提高软件开发过程的能见度

(2)提高开发效率

(3)作为开发人员阶段工作成果和结束标志

(4)记录开发过程的有关信息便于使用与维护；

(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。

20、软件工程三要素之间的关系？

软件质量是软件工程的生命线，软件工程以质量保证为基础。

质量管理促进了过程的改进，创造了许多行之有效的软件开发方法和工具。

软件工程釆用层次化的方法，每个层次都包括过程、方法、工具三要素。

方法支撑过程和工具，过程和工具又促进方法学的研究。

21、说明面向对象的特征？

面向对象的特征是：

(1)对象唯一性。每个对象都有自身唯一的标识，在对象生存期中，其标识不变，不同的对象不能有不同的标识。

(2)分类性。是指将具有一致的数据结构(属性)和行为(操作)的对象抽象成类

(3)继承性。是子类自动共享父类数据结构和方法的机制，这是类之间的一种关系。

(4)多态性。是指相同的操作或函数过程可以作用于多种类型的对象上并获得不同的结果。

22、简述”变换分析”设计步骤。

⑴确定变换中心，逻辑输入和输出。

⑵设计软件结构的顶层和第一层–变换结构。

⑶设计中﹑下层模块，对第一层的模块自顶向下分解。

⑷设计的优化 ：输入部分求精，每个物理输入设置专门模块，以体现系统的外部接口。其它输入模块与转换数据模块都很简单时，可将它们合并。

23、请简述可行性研究有哪些步骤？

复查系统规模和目标；研究当前的系统；导出新系统的高层逻辑模型；重新定义问题；导出和评价可供选择的解决方案；推荐可行的行动方案；草拟开发计划；书写文档并提交审查。

24、请简述黑盒测试和白盒测试有何区别，黑盒测试有哪些常用方法？

黑盒测试和白盒测试的区别：前者基于功能，后者基于结构；

黑盒测试常用方法有：边界值、等价类、因果图、错误推测法等。

25、什么是软件可维护性？常见的软件维护活动有哪几类？

软件可维护性的定义：软件能够被理解、校正、适应及增强功能的容易程度；

常见的软件维护活动有：改正型、适应型、完善型、预防型。

26、CMM全称是什么?分为哪几个级别？

CMM全称是软件能力成熟度模型；

从无序到有序的进化分成5个级别，分别为：

1 初始级；2 可重复级；3 已定义级；4 已管理级；5 优化级。

27、什么是数据字典?简述数据字典与数据流图的关系。

数据字典是关于数据的信息的集合，对数据流程图中的各个元素做完整的定义与说明，是数据流程图的补充工具；

数据字典与数据流图的关系：数据流图和数据字典共同构成系统的逻辑模型，没有数据字典数据流图就不严格，然而没有数据流图数据字典也难于发挥作用。 数据流图和对数据流图中每个元素的精确定义放在一起，才能共同构成系统的规格说明。

28、软件生命周期划分为哪几个阶段？

软件生命周期分为三个时期八个阶段：

软件定义：问题定义、可行性研究；

软件开发：需求分析、概要设计、详细设计、编码、测试；

软件运行：软件维护

29、简述三种面向对象模型的主要功能？

①对象模型：表示了静态的结构化的系统数据性质，描绘了系统的静态结构，从客观世界实体的对象关系角度来描绘对象。

②动态模型：该模型描述了系统的控制结构，它表示了瞬间的、行为化的系统控制性质，它关心的是系统的控制及操作的执行顺序，它从对象的事件和状态的角度出发，表现了对象的交互行为。

③功能模型：表示变化的系统“功能”性质，它指明系统应该“做什么”，因此功能模型更直接的反映了用户对目标系统的要求。

30、详细设计的基本任务是什么？

详细设计的基本任务包括：为每个模块进行详细的算法设计；为模块内的数据结构进行设计；对数据库进行物理设计；其他设计；编写详细设计说明书；评审。

31、黑盒测试和白盒测试有什么区别？

（1）黑盒测试法: 把程序看作一个黑盒子，完全不考虑程序的内部结构和处理过程。它只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当地接收输入数据并产生正确的输出信息，程序运行过程中能否保持外部信息的完整性。黑盒测试又称为功能测试，包含等价类划分法、边界值法、错误推测法、因果图法等。

（2）白盒测试法: 是把程序看成装在一个透明的白盒子里，测试者完全知道程序的结构和处理算法。这种方法按照程序内部的逻辑测试程序，检测程序中的主要执行通路是否都能按预定要求正确工作。白盒测试又称为结构测试，包含逻辑覆盖法和基本路径测试法。

32、什么是软件工程？软件工程面临的问题是什么？

软件工程就是用科学的知识和技术原理来定义，开发，维护软件的一门学科。

软件工程面临的问题：软件的费用，可靠性，可维护性，软件生产率和软件的重用。

33、软件工程三要素之间的关系？

软件质量是软件工程的生命线，软件工程以质量保证为基础。

质量管理促进了过程的改进，创造了许多行之有效的软件开发方法和工具。

软件工程釆用层次化的方法，每个层次都包括过程、方法、工具三要素。

方法支撑过程和工具，过程和工具又促进方法学的研究。

34、简述结构化程序设计方法的基本要点。

(1)采用自顶向下，逐步求精的程序设计方法。

(2)使用三种基本控制结构构造程序，分别是顺序，选择和循环

(3)采用主程序员组的组织形式。

35、顾客在自动柜员机中插入银行卡，执行提款的服务项目；银行职员要求自动柜员机打印所有的提款数据。画出自动柜员机的Use Case图。？

36、简述提高可维护性的方法。

(1)建立明确的软件质量目标；

(2)利用先进的软件开发技术和工具；

(3)建立明确的质量保证工作；

(4)选择可维护的程序设计语言；

(5)改进程序文档。

37、简述文档在软件工程中的作用。

(1)提高软件开发过程的能见度；

(2)提高开发效率；

(3)作为开发人员阶段工作成果和结束标志；

(4)记录开发过程的有关信息便于使用与维护；

(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。

38、什么是软件危机？产生原因是什么？

软件开发技术的进步未能满足发展的要求。在软件开发中遇到的问题找不到解决的办法，问题积累起来，形态尖锐的矛盾，导致了软件危机。

产生原因：

⑴ 软件规模越来越大，结构越来越复杂。

⑵ 软件开发管理困难而复杂。

⑶ 软件包开发费用不断增加。

⑷ 软件开发技术落后。

⑸ 生产方式落后，仍采用手工方式。

⑹ 开发工具落后，生产率提高缓慢。

39、调试的目的是什么?调试有哪些技术手段?

调试则是在进行了成功的测试之后才开始的工作。调试的目的是确定错误的原因和位置，并改正错误，因此调试也称为纠错(Debug)。调试的技术手段有简单的调试方法、归纳法、演绎法和回溯法等。

40、在传统软件工程方法指导下，数据流图和数据字典的关系如何？

数据字典是关于数据的信息的集合，对数据流程图中的各个元素做完整的定义与说明，是数据流程图的补充工具。

数据流图和数据字典共同构成系统的逻辑模型，没有数据字典数据流图就不严格，然而没有数据流图数据字典也难于发挥作用。 数据流图和对数据流图中每个元素的精确定义放在一起，才能共同构成系统的规格说明

**第二份**

1. 简述软件设计的过程。

软件设计是把许多事物和问题抽象起来，并且抽象它们不同层次和角度，是将需求转变为软件陈述的过程，是迭代的过程。

软件设计，要根据需求规格说明书，对整个设计过程进行计划，然后实施具体的设计过程，即“先整体，再局部”，也是不断迭代和精化的过程。

然后需要对生成的设计规格说明书进行评审，启动 质量评价的标准，若未通过评审，需重新修改设计，直至评审通过，确定最后定型的过程本身。进入后续阶段，完成软件设计过程。

2. 简述结构化程序设计方法的基本特点。

1）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。

4） 自顶向下逐步求精。

3. 衡量模块独立性的两个定性标准是什么？这两个标准的定义分别是什么？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？

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）。

4. 什么是黑盒测试法？

黑盒测试法把程序看成一个黑盒子，完全不考虑程序的内部结构和处理过程（2分）。

它只检查程序功能是否能按照规格说明书的规定正常使用（1分），

程序是否能适当地接收输入数据（1分），

产生正确地输出信息（1分）。

5. 如何选择程序设计语言的工程特性

1.系统用户的要求

2.可以使用的编译程序

3.可以得到的软件工具

4.工程规模

5.程序员的知识

6.软件可移植性的要求

7.软件的应用领域

6. 简述面向对象的4个要点的含义。

面向对象的４个要点是：对象分解、数据专有、继承、封装性。（１分）

1）对象分解：认为客观世界是由各种对象组成的。  （１分）

2）数据专有，方法共享：把所有对象都划分成各种对象类(简称为类，class)，每个对象类都定义了一组数据和一组方法。 （１分）

3）继承：按照子类(或称为派生类)与父类(或称为基类)的关系，把若干个对象类组成一个层次结构的系统(也称为类等级)。（１分）

4）封装性：对象彼此之间仅能通过传递消息互相联系。（１分）

7. 软件维护的过程

在软件开发完成交付用户使用后，为了保证软件在一个相当长的时期能够正常运行，就需要对软件进行维护。 软件维护的过程：

（1） 确认维护要求。（1分）

（2） 对于改正性维护申请，评价错误的严重性。对于严重的错误，立即安排人员，分析问题原因，进行"救火"性的紧急维护；对于不严重的错误，根据任务情况和轻重缓急进行统一安排。（2分）

（3） 对于适应性和完善性维护申请，需要确定申请的优先级，然后安排维护工作。并不是所有的完善性维护申请都必须承担，需要考虑商业需要、现有资源、未来发展方向等进行决定。（2分）

8. 软件维护的目的和主要任务是什么？维护的类型有哪些？

软件维护是软件生命周期中最后一个、持续时间最长的阶段。

主要任务是在软件已经交付使用之后，为了改正错误或满足新的需要而修改软件。

目的是提高软件的可靠性、可用性，延长软件的寿命。

维护的类型有：改正性（错误）、适应性（环境）、完善性（功能）、预防性（新产品）

9. 什么是耦合?模块的耦合包括哪些类型？

耦合是对一个软件结构内不同模块之间互连程度的度量。

模块的耦合包括以下几种类型：数据耦合，控制耦合，特征耦合，公共环境耦合，内容   耦合，标记耦合，无耦合/非直接耦合

10. 什么是内聚?模块的内聚包括哪些类型？

内聚标志着一个模块内各个元素彼此结合的紧密程度，它是信息隐蔽和局部化概念的自然扩展

模块的内聚包括以下几种类型：低内聚 — 偶然内聚，逻辑内聚，时间内聚  中内聚 — 过程内聚，通信内聚； 高内聚 — 顺序内聚，功能内聚

11. 什么是编码风格?为什么要强调编码风格?

编码风格包括以下内容：程序内部良好的文档（注释）、数据说明、语句构造、输入输出和效率保障等。

因为编码风格对软件的可读性、可维护性、可靠性、可用性很重要，所以要强调编码风格。

12. 什么是软件危机？软件危机的表现是什么？其产生的原因是什么？

软件发展的第二阶段末期，由于计算机硬件技术的进步，计算机运行速度、容量、可靠性有显著的提高，生产成本显著下降，这为计算机的广泛应用创造了条件。一些复杂的，大型的软件开发项目提出来了，但是，软件开发技术的进步一直未能满足发展的需要。在软件开发中遇到的问题找不到解决方法，使问题积累起来，形成了尖锐的矛盾，因此导致了软件危机。

13. 软件危机表现在以下四个方面：（钱与时间，要求，维护性，可靠性）

（1）经费预算经常突破，完成时间一再拖延。

（2）开发的软件不能满足用户要求

（3）开发的软件维护性差

（4）开发的软件可靠性差

13. 造成软件危机的原因：（规模和结构，管理，经费，技术，工具）

（1）软件的规模越来越大，结构越来越复杂

（2）软件开发管理困难而复杂

（3）软件开发的经费不断增加

（4）软件开发技术落后

（5）生产方式落后开发工具落后，生产率提高缓慢

14. 软件生存周期包括哪些活动？

答：软件生存周期的活动包括：可行性分析和项目开发计划、需求分析、概要设计、详细设计、编码、测试、维护等活动。

15. 请说明 “三个模型” 的定义，以及三者之间的关系。

功能模型是描述系统能做什么，即对系统的功能、性能、接口和界面进行定义。

业务模型是描述系统在何时、何地、由何角色、按什么业务规则去做，以及做的步骤或流程，即对系统的操作流程进行定义。

数据模型是描述系统工作前的数据来自何处，工作中的数据存什么地方，工作后的数据放到何处，以及这些数据之间的关联，即对系统的数据结构进行定义。

功能模型和业务模型是在需求分析时建模，是两个基本点。数据模型是一个中心，在设计时建模。功能模型和业务模型给数据模型提供数据与维护数据，数据模型支持功能模型和业务模型的正常运行。通常，数据模型建模用 Power Designer，ER win 或 Oracle Designer 工具实现；功能模型用功能点列表（或用况图）表示；业务模型用自然语言加上流程图（或顺序图）表示。

16. “面向对象实现” 的目标是什么？

答：“面向对象实现” 的目标是：按照《详细设计说明书》中对数据结构、算法分析和模块实现等方面的设计说明，用面向对象的语言，将这些部件组装起来，分别实现各模块的功能，从而实现目标系统的功能、性能、接口、界面等要求。

**第三份**

**1. 软件危机的表现形式**

* 对软件开发成本和进度估计不准确
* 已完成的软件不符合用户需求
* 软件产品质量差，可靠性得不到保证
* 软件产品可维护性差
* 软件成本在计算机总成本中的比例逐渐变大
* 软件开发生产率提高速度比不上计算机应用速度

**2. 产生软件危机的原因**

* (1)软件是计算机系统的逻辑部件
* (2)软件开发的管理困难
* (3)软件开发费用不断增加
* (4)软件开发技术落后
* (5)生产方式落后
* (6)开发工具落后

**3. 软件生命周期分为哪三个时期？每一时期所完成的基本任务是什么？**

答：软件定义、软件开发、运行维护

**定义时期的任务：**问题定义、可行性研究、需求分析

**开发时期的任务：**软件设计（总体设计、详细设计）；编码与单元测试；综合测试

**维护时期的任务：**维护，使软件持久地满足用户的需要。

**4. 模型以及特点**

**瀑布模型(生存周期模型)：**最经典，需求基本不变，线性的顺序阶段模型

**快速原型模型：**快速获取用户需求

**增量模型：**灵活，允许软件变化

**螺旋模型：**在原型模型和增量模型基础上增加风险分析，用于企业开发

**面向对象：**喷泉模型：以生存周期为主线，具有迭代性和无间隙性

**基于构件的模型：**提高软件重用性和可靠性

**统一过程模型(RUP)：**基于UML的面向对象过程模型

**基于形式化的开发模型：**确保严格、无错

**5. 软件模块设计准则：**

* (1)降低耦合，提高内聚
* (2)模块深度、宽度、扇出扇入应适当
* (3)模块作用范围在控制范围内
* (4)接口设计要简单
* (5)设计功能可能预测并能得到验证的模块
* (6)划分模块规模，保持独立性。

**6. 详细设计的目的和任务**

目的：为软件结构图中每一个模块确定采用的算法和块内数据结构，用某种选定的详细设计工具更清晰的描述，从而在编码阶段可以把这些描述直接翻译成某种程序设计语言书写的源程序。任务：设计出程序的蓝图，以后程序员根据这个蓝图写出实际的代码。

**7. 什么是结构化程序设计？结构化程序设计采用几种控制结构？**

答：结构程序设计：是一种设计程序的技术，它采用自顶向下逐步求精的设计方法和单入口单出口的控制结构。顺序，选择，循环。

**8. 程序设计语言的特点？**

1、名字说明 2、类型说明 3、初始化 4、程序对象的局部性5、程序模块 6、循环控制结构 7、分支控制结构 8、异常处理9、独立编译

**9.软件测试的基本任务是什么？测试与调试的区别是什么？**

答：任务是在软件投入生产性运行之前，尽可能多地发现软件中的错误。

与测试的区别：测试是为了发现软件中的错误，而调试是诊断并改正错误。

**10.可行性研究的步骤？**

(１)复查系统规模和目标。(２)研究目前正在使用的系统。(３)导出新系统的高层逻辑模型。(４)重新定义问题。(５)导出和评价供选择的解法。(６)推荐行动方针。(７)草拟开发计划。(８)书写文档提交审查

**11. 从哪些方面验证软件需求的正确性？**

(1)一致性，即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，指定的需求应该是用现有的硬件技术和软件技术基本上可以实现的。

(4)有效性，必须证明需求是正确有效的，确实能解决用户面对的问题。

**12. 对象的特点？**

(1)以数据为中心。

(2)对象是主动的。

(3)实现了数据封装。

(4)本质上具有并行性。

(5)模块独立性好。

**13. 用面向对象方法开发软件时，通常需要建立哪三种形式的模型？**

(1)描述系统数据结构的对象模型。

(2)描述系统控制结构的动态模型。

(3)描述系统功能的功能模型。

**14. 什么是模块的作用域？什么是模块的控制域？**

模块的作用域定义为受该模块内一个判定影响的所有模块的集合。模块的控制域是这个模块本身以及所有直接或间接从属于它的模块的集合。

**15.简述两种不同集成测试策略的比较？**

答：自顶向下测试方法主要优点是不需要测试驱动程序，能够在测试阶段的早期实现并验证系统的主要功能，而且能在早期发现上层模块的接口错误。自顶向下测试方法的主要缺点是需要存根程序，可能遇到与此相联系的测试困难，低层关键模块中的错误发现较晚，而且用这种方法在早期不能充分展开人力。自底向上测试言法的优缺点与自顶向下测试方法的优缺点刚好相反。

**16. 什么是类？类与对象间是什么关系？**

答：类是对具有相同数据和相同操作的一组相似对象的定义。类是对具有相同属性和行为的一个或多个对象的描述，而对象是某个特定类所描述的一个具体对象。

**17. 面向对象方法特征有哪些？**

(1)客观世界是由各种对象组成的。(2)所有对象都划分成各种对象类，每个对象类都定义了一组数据和一组方法。

(3)按照子类与父类的关系，把若干个对象类组成一个层次结构的系统。

(4)对象彼此之间仅能通过传递消息互相联系。

**18. 可行性研究的目的？应该从哪些方面研究目标系统的可行性？**

答：用最小的代价在尽可能短的时间内确定问题是否能够解决。

(1)技术可行性(2)经济可行性(3)操作可行性(4)法律可行性

**19. 什么是软件工程？构成软件工程的要素是什么？**

**软件工程：**是指导计算机软件开发和维护的工程学科。采用工程的概念、原理、技术和方法来开发与维护软件，把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来。

要素是：方法、工具、过程。

**20.对象有哪几种定义？**

(1)对象是具有相同状态的一组操作的集合。

(2)对象是对属性值和操作的封装。

(3)对象：=<ID，MS，DS，MI>

* ID：是对象的标识或名字
* MS是对象中的操作集合
* DS：是对象的数据结构
* MI ：是对象受理的消息名集合

**21. 类—&—对象间的关系可以概括为哪三种？**

* **归纳关系：**它反映了一个类与若干个互不相容的子类之间的分类关系。
* **组合关系：**它反映了对象之间的构成关系。
* **关联关系：**反映对象之间相互依赖、相互作用的关系。

**22. 简述渐增模型的特点？**

答：(1)在整个开发过程中需求都可能有变化。(2)对应用领域不熟悉。(3)高风险项目(4)用户完全参与到整个开发过程中。(5)使用面向对象语言或第四代语言。

**23. 总体设计的过程如何？**

答：(1)设想供选择的方案。(2)选取合理的方案。(3)推荐最佳方案。(4)功能分解。(5)设计软件结构。(6)数据库设计。(7)制定测试计划(8)书写文档(9)审查和复审

**24. 面向对象方法的主要优点有哪些？**

(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。

**25. 什么是白盒测试?白盒测试主要采用的技术有哪些?**

白盒测试是测试者了解被测程序的内部结构和处理过程,对程序的所有逻辑路径进行测试,在不同点检查程序状态,确定实际状态与预期状态是否一致。

白盒测试主要采用的技术有：路径测试技术和事务处理流程技术，对包含有大量逻辑判断或条件组合的程序采用基于逻辑的测试技术。

**26. 什么是黑盒测试?黑盒测试主要采用的技术有哪些?**

黑盒测试也称为功能测试,它着眼于程序的外部特征,而不考虑程序的内部逻辑结构。测试者把被测程序看成一个黑盒,不用关心程序的内部结构。黑盒测试是在程序接口处进行测试，它只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当地接收输入数据产生正确的输出信息，并且保持外部信息(如数据库或文件)的完整性。

黑盒测试主要采用的技术有:等价分类法、边界值分析法、错误推测法和因果图等技术。

**27. 路径测试技术中几种主要覆盖的含义?举例说明?**

语句覆盖:至少执行程序中所有语句一次。

判定覆盖:使被测程序中的每一个分支至少执行一次。故也称为分支覆盖。

条件覆盖:执行所有可能的穿过程序的控制路流程。

条件组合测试:设计足够的测试用例，使每个判定中的所有可能条件取值组合至少执行一次。

**第四份**

第一章 软件工程

**1.与硬件相比，软件的特点**

(1)不是具体的物理实体，具有抽象性

(2)没有明显的制造过程

(3)存在退化问题

(4)对计算机系统有着不同程度的依赖性

(5)尚未完全摆脱人工的开发方式

(6)软件本身是复杂的

(7)软件成本相当昂贵

**2.为什么软件工程解决了软件危机的各种弊端**

因为软件工程提出是为了解决软件危机所带来的各种弊端。具体的讲，软件工程的目标主要包括以下几点：

(1)使软件的开发成本控制在预计的合理范围之内

(2)使软件产品的各项功能和性能能够满足用户需求

(3)提高软件产品的质量

(4)提高软件产品的可靠性

(5)使生产出来的软件产品易于移植,维护,升级和使用

(6)使软件的开发周期能够控制在预计的合理时间范围内

**3.软件工程的基本原则**

(1)用分阶段的生命周期计划严格管理

(2)坚持进行阶段评审

(3)实行严格的产品控制

(4) 用现代程序设计技术

(5) 结果应能清楚地审查

(6) 开发小组的人员应该少而精。

(7) 承认不断改进软件工程实践的必要性

**4.软件工程的研究内容**

(1)软件开发技术

(2)软件开发过程管理

**5.软件工程的目标**

(1)达到要求的软件功能

(2)取得较好的软件性能

(3)开发出高质量的软件

(4)付出较低的开发成本

(5)需要较低的维护费用

(6)能按时完成开发工作，及时交付使用

**6.软件工程的方法**

(1)结构化方法

(2)面向数据结构方法

(3)面向对象方法

(4)形式化方法

第二章 软件过程

**1.如何理解软件生命周期的内在特征？**

软件产品的生命周期是指从设计该产品的构想开始，到软件需求的确定、软件设计、软件实现、产品测试与验收、投入使用以及产品版本的不断更新，到最终该产品被市场淘汰的全过程。软件生命周期的每个阶段都完成特定的相对独立的任务。合理地划分软件生命周期的各个阶段，使各个阶段之间既相互区别又相互联系，为每个阶段赋予特定的任务

**2.请对比各软件过程模型**

瀑布模型：

瀑布模型是出现得比较早的软件开发模型。在这种模型中，各阶段之间的组织方式就如同瀑布流水一样，逐级下落。开发人员必须在完成前一阶段的任务后，才能开始下一阶段的工作，各个阶段之间通常是按固定顺序连接的，前一阶段的输出往往就是后阶段的输入。

根据瀑布模型的理论，这种模型具有以下几个特点。

1.瀑布模型是一种线性的软件开发模型，回溯性很差。

2.瀑布模型是一种基于里程碑的阶段过程模型。

3.瀑布模型强调软件开发过程的阶段性，每个阶段完成特定的任务。瀑布模型适用于具有以下特征的软件开发项目：

4.在软件开发的过程中，需求不发生或很少发生变化，并且开发人员可以一次性获取到全部需求

5.软件开发人员具有丰富的经验，对软件应用领域很熟悉。

6.软件项目的风险较低。 瀑布模型不具有完善的风险控制机制。

瀑布模型的优点是阶段性强，易于对项目进行管理，缺点是开发过程不灵活，不能适应环境的变化。如果后续阶段中发现前期工作的错误或需求发生了变更，会造成巨大的损失。

原型模型：

原型模型是开发人员为了快速而准确地获取需求经常采用的方法。在初步获取需求后，开发人员会快速地开发一个原型系统。通过对原型系统进行模拟操作，开发人员可以更直观、更全面和更准确地了解用户对待开发系统的各项要求，同时还能挖掘到隐藏的需求。如果开发人员对将采用的开发技术把握不大，也可以采用原型模型进行技术上的尝试，以降低后续开发的风险。原型模型具有以下特点。

1.原型模型主要用于挖掘需求，或是进行某种技术或开发方法的可行性研究。

2.原型系统通常针对软件 开发系统的子功能模块，所以功能相对不完善。

3.由于原型系统功能的局部性以及存在阶段的局部性，在软件开发的实践中，原型模型通常

4.结合其他的软件开发模型共同使用，发挥作用。

原型模型适用于具有以下特征的软件开发项目：

1.对现有的软件系统进行产品升级或功能完善。

2.开发人员与用户之间交流受限，需求获取困难。

3.开发人员对将要采用的技术手段不熟悉或把握性不大。

4.具备快速开发的工具。

原型模型的优点是简单和快速，缺点是需要花费一些额外的成本来构造原型，并且不利于创新。

增量模型：

增量模型是把待开发的软件系统模块化，将每个模块作为一个增量组件，从而分批次地分析、设计、编码和测试这些增量组件。运用增量模型的软件开发过程是递增式的过程。相对于瀑布模型而言，采用增量模型进行开发，开发人员不需要一次性地把整个软件产品提交给用户，而是可以分批次进行提交。增量模型的最大特点就是将待开发的软件系统模块化和组件化。基于这个特点，增量模型具有以下优点。

1.将待开发的软件系统模块化，可以分批次地提交软件产品，使用户可以及时了解软件项目的进展。

2.以组件为单位进行开发降低了软件开发的风险。一个开发周期内的错误不会影响到整个软件系统。

3.开发顺序灵活。

增量模型的缺点是要求待开发的软件系统可以被模块化。如果待开发的软件系统很难被模块化，那么将会给增量开发带来很多麻烦。

增量模型适用于具有以下特征的软件开发项目。

1.软件产品可以分批次地进行交付。

2.待开发的软件系统能够被模块化。

3.软件开发人员对应用领域不熟悉，难以一次性地进行系统开发。

4.项目管理人员把握全局的水平较高。

螺旋模型：

螺旋模型是一种用于风险较大的大型软件项目开发的过程模型。它把开发过程分为制定计划、风险分析、实施工程和客户评估4种活动。制定计划就是要确定软件系统的目标，了解各种资源限制，并选定合适的开发方案。风险分析旨在对所选方案进行评价，识别潜在的风险，并制定消除风险的机制。实施工程的活动中渗透了瀑布模型的各个阶段，开发人员对下一版本的软件产品进行开发和验证。客户评估是获取客户意见的重要活动。

螺旋模型适应于风险较大的大型软件项目的开发。它的优点是将风险分析扩展到各个阶段中，大幅度降低了软件开发的风险。但是这种模型的控制和管理较为复杂，可操作性不强，对项目管理人员的要求较高。

V模型

**3.在统一软件开发过程模型中,核心工作流程包括哪些?**

核心工作流程：业务建模，需求分析，分析设计，实现，测试和部署

4.软件生命周期的过程

(1)可行性研究(2)需求分析(3)软件设计

(4)编码(5)软件测试(6)软件维护

5.简述软件生命周期及其每阶段的主要任务

6个阶段:可行性研究、需求分析、软件设计、编码、测试、维护。

可行性研究：弄清楚“要解决的问题是什么”

需求分析：确定“软件系统必须做什么”

软件设计：确定软件的结构由哪些模块组成

编码：写成以语言表示的“源程序”

测试：尽可能多地发现并排除软件中潜藏出错误

维护：更新功能，升级版本，以保持软件产品正常运行

第三章 可行性研究及需求分析

**1.可行性研究的内容**

(1)战略可行性

(2)操作可行性

(3)计划可行性

(4)技术可行性

(5)社会可行性

(6)市场可行性

(7)经济可行性

(8)风险可行性

**2.需求分析的原则与常用方法**

原则：

首先，需求分析是一个过程，它应该贯穿于系统的整个生命周期中，而不是仅仅属于软件生命周期早期的一项工作。

其次，需求分析应该是一个迭代的过程。由于市场环境的易变性以及用户本身对于新系统要求的模糊性，需求往往很难一步到位。通常情况下，需求是随着项目的深入而不断变化的。所以需求分析的过程还应该是一个迭代的过程。

此外，为了方便评审和后续的设计，需求的表述应该具体、清晰，并且是可测量的、可实现的。最好能够对需求进行适当的量化。

方法：

1.功能分解方法

2.结构化分析方法

3.信息建模方法

4.面向对象的分析方法

**3.可行性研究步骤**

(1)明确系统目标

(2)分析研究现行系统

(3)设计新系统的高层逻辑模型

(4)获得并比较可行的方案

(5)撰写可行性研究报告

第七章 软件设计

**1.软件设计原则**

(1)模块化(2)抽象化(3)逐步求精(4)信息隐藏

(5)复用性设计(6)灵活性设计(7)模块独立性

第十章 面向对象设计

**1.对象：**

类的一个实例

**2.类**

类是一组具有相同属性、操作、关系和语义的对象的描述；

**3.实例**

实例是实体的具体的例子,关系是用来连接实体的。

继承：继承就是子类继承父类的特征和行为，是多态的一个体现

多态：同一个对象，调用不同类的实例，从而得到不同的结果

**4.简述类在建模中的主要用途**

类图主要用在面向对象软件开发的分析和设计阶段，描述系统的静态结构。

**5. 简述面向对象(模块化)的设计原则**

(1)模块化(2)抽象化(3)信息隐藏

(4)低耦合(5)高内聚(6)复用性

第十四章 软件维护

**1.软件维护的分类**

纠错性维护：为了识别并纠正软件产品中所潜藏的错误,改正软件性能上的缺陷进行的维护

适用性维护：为了使软件产品适应软硬件环境的变更而进行的维护

完善性维护：软件维护的主要部分，是针对用户对软件产品提出的新需求而进行的维护

预防性维护：主要采用先进的软件工程方法对已过时的软件系统进行重新设计、编码和测试，达到结构上的更新

其他补充：

**1.在模块化设计中必须遵守的设计原则是什么？**为了实现有效的模块化，采用的启发设计策略是什么？

在模块化设计过程中，必须遵循的设计原则:

抽象与逐步求精、信息隐藏、模块化、高内聚、低耦合。

按照模块化设计原则，相应的启发式设计策略(6条):

(1)改造程序结构图，降低耦合度，提高内聚度

(2)避免高扇出，并随着深度的增加，力求高扇入

(3)模块的影响范围应限制在该模块的控制范围内

(4)降低模块接口的复杂程度和冗余程度，提高一致性

(5)模块的功能应是可预测的，避免对模块施加过多的限制

(6)尽可能设计单入口和单出口的模块

**2.什么是需求规约？**简述需求规约的基本性质。

答：需求规约是一个软件项/产品/系统所有需求陈述的正式文档，它表达了一个软件产品/系统的概念模型。需求规约一般需要满足一下4个基本性质：

1.重要性和稳定性程度：按需求的重要性和稳定性，对需求进

行分级；

2.可修改性：在不影响其他需求的前提下可容易修改一个单一

需求；

3.完整性：设备被遗漏的需求；

4.一致性：不存在互斥的需求。

**3.什么是模块耦合？**简述常用的模块耦合类型及其设计原则。

答：模块耦合：是指不同模块之间相互依赖程度的度量；

几中常见模块耦合类型为：内容耦合、公共耦合、控制耦合、标记耦合、数据耦合等；

设计原则：如果模块间必须存在耦合，就尽量使用数据耦合，少用控制耦合，限制公共耦合，避免内容耦合。

**4.简述软件开发与软件测试之间的关系**

开发和测试是一个有机的整体，在产品的发布之前，开发和测试是循环进行的，测出的缺陷要经开发人员修改后继续测试。在开发的同时测试经理开始编写测试用例，测试文档要参考开发文档，所以开发和测试是不可分割的，少了任何一个都不能开发出产品。

**5.什么是黑盒测试法，常用的黑盒测试方法有哪些?**

在测试中，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性，着眼于程序外部结构，不考虑内部逻辑结构，主要针对软件界面和软件功能进行测试。

常用的黑盒测试技术有：等价类划分法、边界值分析法、错误推测法、因果图法

**6.什么是白盒测试法，常用的白盒测试方法有哪些?**

关注软件产品的内部细节和逻辑结构，把被测的程序看成一个透明的盒子

常用的白盒测试技术有：代码检查法、逻辑覆盖测试、基本路径测试

**7.UML统一建模语言有哪几种模型？各种模型的主要功能是什么**

功能模型: 从用户的角度展示系统的功能，包括用例图。

对象模型: 采用对象，属性，操作，关联等概念展示系统的结构和基础，包括类图、对象图、包图。

动态模型: 展现系统的内部行为。 包括序列图，活动图，状态图。