概述

填空题

1. 在软件开发的结构化方法中，采用的主要技术是 SA，即  结构化需求分析    和 SD，即     结构化设计    。
2. 喷泉模型                    模型比较适用于面向对象的开发方法。
3. SWEBOK V3 中，软件工程知识体细分为“  软件工程教育需求    ”和“    软件工程实践   ” 两大类。
4. 软件代理一般具有  自治性   、    开放性       、       反应能力      等特性。
5. 增量模型              模型假设需求可以分段，成为一系列增量产品，每一增量可以分别开发。
6. UML 的三个基本构造块是        事物          、         关系         和         图         。
7. 软件工程是用工程科学的知识和技术原理来      定义       、     开发        和       维护            软件的一门学科
8. 软件工程是一门综合性的交叉学科，它涉及计算机学科，   工程   学科、   管理       学科和     数学   学科
9. 软件工程研究的主要内容是      方法        、           过程        和          工具             等三个方面。
10. 数据流图是描述数据在软件中流动和被处理的过程，是软件模型的一种图示，它一般包 括 4 种图形符号：       加工                、         外部实体               、        数据流向                、          数据存储            。
11. 面向对象方法               是将数据和对数据的操作紧密地结合起来的方法，这是与传统结构化 方法的主要区别。
12. 由于软件生产的复杂性和高成本，使大型软件生产出现了很多问题，即出现       软件危机         ， 软件工程正是为了克服它而提出的一种概念及相关方法和技术。

问答题

1. 什么是软件危机？软件危机是指计算机软件开发和维护能够中所遇到的一系列严重问题。
2. 什么是软件？软件是计算机系统中与硬件相互依存的另一部分，包括程序、相关数据及其说明文档。
3. 什么是软件工程？软件工程是用工程科学的知识和技术原理来定义、开发和维护软件的一门学科。
4. 面向过程的软件工程方法的基本活动有哪些？结构化需求分析、结构化设计、结构化编程、结构化测试、结构化维护
5. 软件工程学包括哪些内容？软件工程学包括软件开发技术和软件工程管理两方面的内容。其中软件开发方法学和软件工程环境属于软件开发技术的内容。软件工程经济学属于软件工程管理的内容。
6. 常见的软件开发模型有哪些？瀑布模型、V模型、原型模型、增量式模型、喷泉模型、智能模型、敏捷生存期模型、螺旋模型。
7. 面向对象的方法的基本活动有哪些？面向对象分析、面向对象设计、面向对象编程、面向对象测试、面向对象维护
8. 软件项目过程改进的线路图是什么？启动、诊断、建立、实施、评估
9. 软件危机的表现有哪些？对软件开发成本和进度的估计常常准确、开发的软件不能满足用户要求、软件产品的质量低、开发的软件可维护性差、没有适当的文档资料、软件成本不断提高、软件开发生产率的提高赶不上硬件的发展和人们需求的增长
10. 软件工程“三段论”是指什么？一段论是“软件项目管理”、二段论是“软件项目开发”、三段论是“软件过程改进”
11. 软件工程的各种方法在分析方法上有什么区别？、面向过程的方法：利用面向过程程序语言和程序设计流程图、面向对象的方法：系统抽象为对象集合、面向构件的方法：系统划分为功能模块，逐个实现，再连接起来、面向代理的方法：系统抽象为具有拟人化的代理集合
12. 软件项目开发的线路图是什么？、需求分析、概要设计、详细设计、编码、测试、产品交付、维护
13. 面向构件的软件工程包括哪些活动？、构件识别、构件适配、构件组装、构件演化
14. 软件工程的最终目的是什么？、软件工程的最终目的是摆脱手工生产软件的状况，逐步实现软件开发和维护的自动化。
15. UML中的基本关系有哪些？、关联关系、依赖关系、泛化关系、实现关系
16. 软件项目管理的线路图是什么？、项目初始、项目计划、项目执行与控制、项目结束
17. 软件工程研究的主要内容是什么？、软件工程研究的主要内容是方法、过程和工具。

需求

1. 用例需求分析方法采用的是一种结构化的情景分析方法，即是一种基于场景建模的方法。（F ）
2. 面向对象分析方法认为系统是对象的集合，是以功能和数据为基础的。（F ）
3. 结构化分析方法适合于数据处理类型软件的需求分析。（T ）

填空题

1. 最常见的实体关系图的表示法是（  Peter Chen  ）表示法和（ James Martin ）表示法。
2. 结构化分析方法是面向  数据流 进行需求分析的方法。结构化分析方法使用   数据流图   等来描述。
3. 在需求分析中，可从有关问题的简述中提取组成数据流图的基本成分。通常问题简述中 的动词短语将成为数据流图中的\_\_\_处理\_\_\_成分。
4. 面向对象的需求分析中常用的 UML 图示有    用例图     、          顺序图            、和       活动图    等。
5. 分析模型在系统级描述和（  设计模型  ）之间建立了桥梁。

问答题

1. 按照IEEE1998,需求可以分为哪几类？功能需求、性能需求、质量属性、对外接口、约束
2. 建立用例图需求模型的基本步骤是什么？分析目标、确定角色（参与者）、确定需求用例、分解细化用例、用例描述
3. 可行性分析的内容有哪些？技术可行性、经济可行性、运行可行性、操作可行性、法律可行性
4. 需求变更管理活动的主要任务有哪些？、确定变更控制过程、建立变更控制委员会、进行变更影响分析、跟踪变更影响的产品、建立基准和控制版本、维护变更的历史记录、跟踪每项需求的状态、衡量需求稳定性
5. 需求获取需要执行哪些活动？建立需求分析小组，收集资料、确定系统的整体目标和系统的工作范围、对用户进行访谈和调研、分析整理收集到的用户需求、确认调研需求
6. 需求验证包括哪些内容？需求正确性、需求一致性、需求完整性、需求可行性、需求必要性、需求可验证性、需求可追踪性、验证最后是否经过了签字确认
7. 把需求工程的管理划分为哪几个独立的过程？需求获取、需求分析、需求规格说明编写、需求验证、需求变更
8. 数据流图的基本元素是什么？过程、实体、数据流和数据存储
9. 按照层次，需求可以分为哪几层？按照层次，需求可以分为：业务需求、用户需求、功能需求和软件需求规格说明。

概要设计

1. 数据存储和数据流都是（ 数据），仅仅是所处的状态不同。
2. 下列关于软件设计准则的描述，错误的是（ 使摸块的作用域在该模块的控制域外   ）。
3. 面向数据流的软件设计方法中，一般是把数据流图中的数据流分为（ 交换流和事务流 ）两种流，再将数据流图映射为软件结构。
4. 面向数据流的设计方法把（  信息流）映射成软件结构。
5. 软件设计是软件工程的重要阶段，是一个把软件需求转换为软件代码的过程（F）

填空题

1. 结构化程序设计方法的主要原则可以概括为（自顶向下，逐步求精，模块化）。
2. 软件模块设计，包括模块划分， （模块的层次结构） ，模块的调用关系，每个模块的功能 等等
3. 设计模型是从分析模型转化而来的，主要包括四类模型：（ 体系结构设计模型 ）、数据设 计模型、接口设计模型、构件设计模型。
4. 数据字典包括（    数据项    ）、（    数据流     ） 、 数据储存和基本加工.
5. （设计模型 ）把已确定的软件需求转换成特定形式的设计表示，使其得以实现。
6. 模块层次图和模块结构图是（ 模块划分 ）的重要方法
7. UML 设计中主要采用的图示有（类图   ）、（对象图）、（包图）等
8. 面向对象设计的主要特点是建立了四个非常重要的软件设计概念：抽象性、（  信息隐蔽性）、功能独立性和模块化。
9. 高内聚、松耦合是 （    模块设计       ）的基本原则
10. 数据库的设计一般要进行三个方面的设计：（概念设计）、（逻辑设计）、（物理设计）
11. C/S、B/S、SOA  、BMP 等都是不同的（  体系结构 ）

问答题

1. 你知道内聚有哪些？分别是什么含义？偶然内聚：块内各个任务之间没有有意义的联系。逻辑内聚：一个模块完成的任务在逻辑上相同或相似。时间内聚：一个模块内包含的任务需要在同一时间执行。过程内聚：一个模块内包含的各个处理元素是相关的，而且必须按固定的次序执行。通信内聚：一个模块内的各个处理元素引用共同的数据。顺序内聚：若一个模块的各处理元素关系密切，必须按规定的处理次序执行功能内聚：一个模块仅完成单一功能

2. 你知道有哪几种耦合类型？分别是什么含义？独立耦合：没有直接联系的模块。数据耦合：两个模块彼此交换数据。控制耦合：两个模块传递的不是数据参数而是控制参数。公共耦合：若干个模块对同一数据区进行存取操作。内容耦合：一个模块直接访问另一个模块的内部代码或数据

3. 高质量设计的特征？可扩展性、模块独立性、错误预防和错误处理、代码重用性设计、友好的人机交互界面

4. 你知道的体系结构有哪些？H/T、C/S、B/S、多层架构、SOA面向服务结构、BPM面向工作流引擎架构

详细设计

1. 结构化程序流程图中一般包括 3 种基本结构，下述结构中（   嵌套结构）不属于其基本结构。
2. JSD 设计方法是由 Jackson 所提出的，它是一种面向（数据结构 ）的软件设计方法
3. 程序流程图中的箭头代表的是（ 控制流）。
4. 软件设计中，可应用于详细设计的工具有（程序流程图、PAD 图、N-S 图和伪代码 ）
5. 伪码作为详细设计的工具，缺点在于（ 不如其他图形工具直观，描述复杂的条件组合与动作间的对应关系不够明了  ）。
6. 在详细设计阶段，一种二维树型结构并可自动生成程序代码的描述工具是(  PAD )。
7. 伪码又称为过程设计语言 PDL，一种典型的 PDL 是仿照（PASCAL 语言  ）编写的。
8. PAD 是一种改进的图形描述方式， 优点是能够反映和描述自顶向下的历史和过程。（T）

填空题

1. 在详细设计阶段，除了对模块内的算法进行设计，还应对模块内的（数据结构）进行设 计。
2. 面向数据结构的设计方法主要包括（ JSD 设计方法）和（ Warnier 设计方法） 。
3. 软件详细设计需要设计人员对每个设计模块进行描述，确定所使用的（  算法）、接口细 节和输入、输出数据等。
4. PDL 又称（伪代码 ），它是一种非形式化的比较灵活的语言
5. 结构化设计方法与结构化分析方法一样，采用（逐步求精 ）技术。结构化设计方法与结 构化分析方法相结合，依数据流图设计程序的结构。
6. 软件中详细设计一般是在（ 概要设计）基础上才能实施，它们一起构成了软件设计的全 部内容。
7. 软件的详细设计可采用图形、（ 表格工具）和过程设计语言等形式的描述工具表示模块 的处理过程。
8. 在 Warnier 方法中，采用（ Warnier 图）表示数据结构和程序结构。

编码

1. 编码高效率原则包括提高运行效率，提高储存效率和提高(输入输出效率 )
2. 下面是一段求最大值的程序，其中 datalist 是数据表，n 是数据表的长度，则其 McCabe 环路复杂性为 （ 3 ） int GetMax(int n, int datalist[ ])  {        int k=0;                 for ( int j=1; j<n; j++ )  if ( datalist[j] > datalist[k] )  k=j;            return k;  }
3. 应该尽量使用机器语言编写代码，提高程序运行效率，而减少高级语言的使用（F）

填空题

1. 任何程序都可由（顺序）、 （选择）和（重复）等三种基本控制结构构造。这三种基本控制结 构的共同点是（ 单入口）和（单出口）。
2. 可以将程序设计语言分为（第一代语言）、（第二代语言）、（第三代语言）、（第四代语言）和（第五代语言）五类。
3. 在软件编码过程中，可以采用（自顶向下）、（自底向上）、（自顶向下和自底向上相结合）以及（ 线程模式）等几种编码策略。

测试

1. 以下（ 边界值分析）不属于白盒测试技术
2. 通过程序设计的控制结构导出测试用例的测试方法是（白盒测试 ）
3. 单元测试又称为（ 模块测试   ），可以用白盒法也可以采用黑盒法测试。
4. 在测试中，下列说法错误的是（   测试是为了表明程序的正确性  ）。
5. （  集成测试  ）方法需要考察模块间的接口和各个模块之间的关系。
6. 回归测试是纠错性维护中最常运用的方法。（F）
7. 回归测试是指在单元测试基础上将所有模块按照设计要求组装成一个完整的系统进行 的测试（F）

填空题

1. 在白盒测试中，对程序的语句逻辑有6种覆盖技术，其中发现错误能力最强的技术是（路径覆盖技术）。
2. 集成测试的策略主要有（大爆炸）、（自顶向下的策略）、（自底向上的策略）、三明治集成测试。
3. 黑盒测试主要针对功能进行的测试，等价类划分、（ 场景法）、错误猜测和因果图法等都是采用黑盒技术设计测试用例的方法。
4. 边界值分析是将测试边界情况作为重点目标，选取正好等于、刚刚大于或刚刚小于边界 值的测试数据。如果输入输出域是一个有序集合，则集合的第一个元素和（ 最后）元素应该 作为测试用例的数据元素。
5. 若有一个计算类程序，它的输入量只有一个 X，其范围是[-1.0,1.0]。现在设计一组测试用 例，X 输入为-1.001,-1.0,1.0,1.001，则设计这组测试用例的方法是（边界值分析法）。

单元测试主要测试模块的5个基本特征（模块接口）、 （局部数据结构）、(重要的执行路径)、( 错误处理) 和(边界条件 )。

1. 从是否需要执行被测软件的角度，软件测试方法一般可分为两大类，即（静态测试）和 （动态测试）方法。
2. 逻辑覆盖包括：（语句覆盖）、（判定覆盖）、（条件覆盖）、（判定/条件覆盖）、条件组合覆 盖和路径覆盖等。

问答题

1. 探索性测试分为哪几种？自由式探索测试、基于场景的探索性测试、基于策略的探索性测试、基于反馈的探索性测试
2. 按照测试基本，测试可以分为哪几个阶段？单元测试、集成测试、系统测试、验收测试
3. 集成测试的方法有哪些？大爆炸集成、自顶向下集成、自底向上集成、三明治集成、冒烟测试
4. 按照测试软件是否被执行，可以分为静态测试和动态测试，静态测试和动态测试常用的方法有哪些？静态测试常用的方法有：文档审查、代码检查、技术评审；动态测试常用的方法有白盒测试、黑盒测试和灰盒测试
5. 白盒测试的覆盖技术有哪些?语句覆盖、判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖、路劲覆盖
6. 黑盒测试的常用方法有哪些？边界值分析法、等价类划分法、规范导出法、错误猜测法、基于故障的测试方法、因果图法、决策表法、场景法

维护

1. 软件按照设计的要求，在规定时间和条件下达到不出故障、持续运行要求的质量特性称 为（  可靠性 ）。
2. 软件维护是软件生命周期中的固有阶段，一般认为，各种不同的软件维护中以（预防性维护  ）维护所占的维护量最小
3. 产生软件维护的副作用，是指（因修改软件而造成的错误）。
4. 诊断和消除程序在使用过程中发生错误的过程称为（校正性维护）
5. 度量软件的可维护性可以包括很多方面，下列（  程序的无错误性 ）不在措施之列。
6. 下面哪个不是交付过程的文档（开发合同    ）。
7. 为适应软件运行环境的变化而修改软件的活动称为（适应性维护）
8. 对于软件的 （可维护性  ），有一种简单的面向时间的度量，叫做平均变更等待时间 MTTC （Mean Time To Change)。 这个时间包括开始分析变更要求、设计合适的修改、实现    变更 并测试它、以及把这种变更发送给所有的用户
9. 在软件生存期的维护阶段，继续诊断和修正错误的过程称为（改正性维护）。

判断

1. 适应性维护是针对系统在运行过程中暴露出来的缺陷和错误而进行的，主要是修改错误。F
2. 可维护性、可使用性、可靠性是衡量软件质量的几个主要质量特性，其中软件的可使用 性是软件各个开发阶段的关键目标 F
3. 完善性维护主要是为了改善系统的某一方面而进行的变更，可能这种变更是因为出现错 误而进行的变更。F
4. 可测试性表明验证程序正确性的容易程度，程序越简单，验证其正确性越容易。T
5. 软件项目交付时要给用户提供必要的文档 T
6. 用户手册不仅要提供系统的使用方法，还需提供系统功能的详细实现方法 F
7. 回归测试是纠错性维护中最常运用的方法。F

填空

1. 如果软件是可测试的、可理解的、可修改的、可移植的、可靠的、有效的、可用的，则 软件一定是可（维护）的。
2. 软件维护的类型主要包括（纠错性维护）、（适应性维护）、（完善性维护）和（预防性维护）等。
3. （系统管理员手册 ）是为系统管理员准备的文档资料。
4. 预防性维护也称为（软件再工程）。
5. 产品交付需要完成的主要任务是（提交产品）和（进行培训） 。
6. 软件的逆向工程是一个恢复（设计结果 ）的过程
7. 软件的可维护性是指纠正软件系统出现的（ 错误或者缺陷）以满足新的要求而进行修 改、扩充或压缩的容易程度。
8. （验收测试 ）是交付使用前的最后一次检查，也是软件投入运行之前保证可维护性的最后机会。
9. 一个可移植的程序应该具有结构良好、灵活、（不依赖于某一具体计算机或者操作系统） 的性能。
10. （ 项目验收）是项目移交的前提，移交时，项目移交方和项目接收方将在项目移交报告上签字，形成项目移交报告。
11. 一个产品的使用者有两种类型：一种是（用户），另外一种是（系统管理员）
12. （第三方测试）是由独立于软件开发者和用户的第三方所进行的测试，旨在对被测软件进 行质量认证。
13. 当一个系统已经在实际环境中投入使用了，可以进行正常的操作，我们就说系统开发完 成了，而以后对系统变更所做的任何工作，称为（维护 ）。