**软件工程概述 第一章**
﹑软件产品的特性是什么？1
产品特性：
⑴是一种逻辑产品，与物质产品有很大的区别。
⑵软件产品的生产主要是研制，生产成本主要在开发和研制，开发研制完成后，通过复
制就产生了大量软件产品。
⑶软件产品不会用坏，不存在磨损，消耗。
的"定做"⑷生产主要是脑力劳动，还末完全摆脱手工开发方式，大部分产品是
⑸开发软件的费用不断增加，致使生产成本相当昂贵。
﹑软件生产有几个阶段？各有何特征？2
⑴程序设计时代：这个阶段生产方式是个体劳动，使用的生产工具是机器语言，汇编语
言。
⑵程序系统时代：这个阶段生产方式是小集团合作生产，使用的生产工具是高级语言，
开发方法仍依靠个人技巧，但开始提出结构化方法。
⑶软件工程时代：这个阶段生产方式是工程化的生产，使用数据库﹑开发工具﹑开发环
境﹑网络﹑分布式﹑面向对象技术来开发软件。
﹑什么是软件危机？主要表现在哪些方面？3
问在软件开发中遇到的问题找不到解决的办法，软件开发技术的进步未能满足发展的要求。
题积累起来，形态尖锐的矛盾，导致了软件危机。
表现方面：
用户对开发出的软件很难满意。(1)
软件产品的质量往往靠不住。(2)
一般软件很难维护。(3)
软件生产效率很低。(4)
软件开发成本越来越大。(5)
软件成本与开发进度难以估计。(6)
软件技术的发展远远满足不了计算机应用的普及与深入的需要。(7)
﹑什么是软件工程？它目标和内容是什么？4
软件工程：就是用科学的知识工程和技术原理来定义，开发，维护软件的一门学科。
开发的软件易于取得较好的性能；软件工程的目标：付出较低开发成本；达到要求的功能；
只需较低的维护费用；能按时完成开发任务，及时交付使用；开发移植；
的软件可靠性高。
软件开开发技术主要研究：研究内容包括开发技术和开发管理两个方面。软件工程的内容：
发方法，开发过程，开发工具和环境。开发管理主要研究：软件管理学，
软件经济学，软件心理学。
5 ﹑软件工程面临的问题是什么？
软件生产率和软件的重用。可维护性，可靠性，软件的费用，软件工程需要解决的问题：
﹑什么是软件生存周期？它有哪几个活动？6
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。
﹑什么是软件生存周期模型？有哪些主要模型？7
对软件开发提供强有力描述软件开发过程中各种活动如何执行的模型。生存周期模型：
是为参与开发的人员提供帮助和指导，为开发过程中的活动提供统一的政策保证，的支持，
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。
- 1 -

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。
﹑什么是软件开发方法？有哪些主要方法？8
通过使用成功的软件使用早已定义好的技术集及符号表示习惯来组织软件生产的过程。
软件开发方法是开发出符合用户需求的高质量的软件。在规定的投资和时间内，开发方法，
克服软件危机的重要方面之一，对软件工程及软件包产业的发展起了不可估量的作用。
﹚，面向对象开发方法。VDM方法，维也纳开发方法﹙JACKSON主要有：结构化方法，
、软件工程方法学包含9 个要素分别是什么？3
分别是：方法、工具和过程。
的问题；”怎样做“方法是完成软件开发的各项任务的技术方法，回答 
工具是为运用方法而提供的自动的或半自动的软件工程支撑环境； 
过程是为了获得高质量的软件所需要完成的一系列任务的框架，它规定了完成各项 
任务的工作步骤。
、什么是软件过程？10
为了获得高质量软件所需要完成的一系列任务的框架，规定了完成各项任务的工作步
在过程结束时将把输并且使用适当的资源，在完成开发任务时必须进行一些开发活动，骤。
入转化为输出。
、瀑布模型的特点及优点？11
特点：
各个阶段的顺序性和依赖性；(1) 
划分逻辑设计与物理设计，尽可能推迟程序的物理实现；(2) 
每个阶段必须完成规定的文档，对其中问题通过复审及早发现，及早解决。(3) 
优点：
；可强迫开发人员采用规范的方法（例如，结构化技术） 
严格地规定了每个阶段必须提交的文档； 
要求每个阶段交出的所有产品都必须经过质量保证小组的仔细验证。 
、螺旋模型的优点？12
对可选方案和约束条件的强调有利于已有软件的重用，也有助于把软件质量作为软 
件开发的一个重要目标；
减少了过多测试（浪费资金）或测试不足（产品故障多）所带来的风险； 
 在螺旋模型中维护只是模型的另一个周期，在维护和开发之间并没有本质区别。
螺旋模型主要适用于内部开发的大规模软件项目。项目越大，风险也越大，因此，
进行风险分析的必要性也越大。此外，只有内部开发的项目，才能在风险过大时方
便地中止项目。
、快速原型的分类、特点及使用策略？13
分类：探索型、实验型、进化型
以便使用户和开发者在尽快向用户提供一个可在计算机上运行的目标系统的模型，特点：
”做什么“目标系统应该 这个问题上尽可能快地达成共识。
；”快速“快速原型应该具备的第一个特点是 
试用—修改“。原型的”容易修改“快速原型应该具备的第二个特点是 过程可”反馈—
能重复多遍。
使用策略：废弃策略和追加策略。
、软件工程的目标？14
开发的软件易于移植；只需较低的达到要求的功能；取得较好的性能；付出较低开发成本；
- 2 -

**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。
、软件开发技术主要研究内容？15
软件开发方法，开发过程，开发工具和环境。
、软件开发管理主要研究内容？16
软件管理学，软件经济学，软件心理学。
、快速原型开发的步骤？17
快速开发、需求、构造原形、原形、运行原形、评价原形、修改意见。
元，假设系统的投资额为2O0O 年内每年可节省人民币5 、某计算机系统投入使用后、18
％。试计算投资回收期和纯收入。12 元，年利率为5000** **可行性研究 第二章
? 、可行性研究的任务是什么1
约束和限制。目标，初步确定项目的规模，首先需要进行概要的分析研究，可行研究的任务：
分析员再进行简要的需求分析，抽象出项目的逻辑结构，建立逻辑模型。从逻辑模型出发，
对每种解决方法都要研究它的可行性。探索出若干种可供选择的解决方法，经过压缩的设计，
主要从四个方面考虑：
开发的风险即设计出的系统能否达到要求的功能和 一般要考虑的情况包括技术可行性：.⑴
相关技术的发展是否支持;资源的有效性;性能;
确定要开发的项目是否值得,经济可行性：进行开发成本的估算以及了解取得效益的评估.⑵
投资。
法律可行性：要开发的项目是否存在任何侵权问题。.⑶
现有管理制度﹑人员素质﹑操作方式是否,管理可行性：运行方式在用户组织内是否可行(4).
可行。
? 、可行性研究有哪些步骤2
确定项目规模和目标；.⑴
研究正在运行的系统；收集﹑研究﹑分析现有系统的文档资料，实地考察系统访问有关.⑵
人员，然后描绘现有系统的高层系统流程图。
.⑶建立新系统的高层逻辑模型；使用数据流图和数据字典描述数据在系统中的流动和处理
情况。
导出和评价各种方案；导出若干较高层次的物理解决方法，根据技术可行性﹑经济可行.⑷
性﹑社会可行性进行评估，得到可行的解决方法。
推荐可行方案；进行成本～效益分析，决定该项目是否值得开发，若值得开发那么解决.⑸
方案是什么，并且说明该方案可行的原因和理由。
编写可行性研究报告；将上述可行性过程的结果写成相应文档，即可行性研究报告。.⑹
、可行性研究过程包括哪些？3
复查系统规模和目标1.
研究目前正在使用的系统2.** 
- 3 -

导出新系统的高层逻辑模型3.
进一步定义问题4.
导出和评价供选择的解法5.
推荐行动方针6.
草拟开发计划7.
书写文档提交审查8.
、什么是数据流图？其作用是什么？4
DFD数据流图（：以图形的方式描述数据在系统中流动和处理的逻辑过程。只反映系统必）
须完成的逻辑功能，是一种功能模型。
、什么是数据字典？其作用是什么？共有哪些条目？5
）用来定义数据库流图中的各个成分的具体含义。有以下四类条目：数据流，DD数据字典（
数据项，数据存储，基本加工。
效益分析可用哪些指标进行度量？--、成本6
通常把建立系统若干年后能取得的收益折算成现在的价值和开发系统所需的:投资回收率.⑴
费用进行比较得出投资回收率。
投资回收期：就是使累计的经济效益等于最初的投资费用所需的时间。.⑵
纯收入：整个生存周期之内的累计经济效益（折成现在值）与投资之差。.⑶
、某图书管理系统有以下功能：7
对于第一次借书的读者，若有效，系统首先检查借书证是否有效，输入读者借书证。⑴借书：
本，若已达10在借书证上建立档案。否则，查阅借书文件，检查该读者所借图书是否超过
10本，办理借书（检查库存，修改库存目录并将读者借书情况录入借书文10，拒借，末达
）件。
个月）作罚款处3⑵还书：从借书文件中读出与读者有关的记录，查阅所借日期，如超期（
理。否则，修改库存目录与借书文件。
⑶查询：通过借书文件，库存目录文件查询读者情况﹑图书借阅及库存情况，打印统计表。
解：
- 4 -

- 5 -

、某银行的计算机储蓄系统功能是：将储户的存户填写的存款单或存款单输入系统，如果8
并打印出存款单系统记录存款人姓名﹑住址﹑存款类型﹑存款日期﹑利率等信息，是存款，
描绘该功能的需求，并建IDEF0，DFD给储户；如果是取款，系统计算清单给储户。请用
立相应数据字典。
- 6 -

一个考务处理系统的要求如下，试画出该系统的数据流图。
、9
对考生送来的报名表进行检查； ）1（
并将汇总后的考对合格的报名表编好准考证号码后将准考证送给考生， ）2（
生名单送给阅卷站；
（并根据考试中心指定的合格标准审定对阅卷站送来的成绩表进行检查， ）3
合格者；
不合格标志），送给考生；∕填写考生通知单（内容包含考试成绩及合格 ）4（
）5（按地区、年龄、文化程度、职业、考试级别等进行成绩分类统计及试题
难度分析，产生统计分析表。
）顶层数据流图1（
层数据流图1第 ）2（
层数据流图2第 ）3（
子图：1加工
- 7 -

子图：2加工
、系统流程图10
是概括地描绘物理系统的传统工具，表达的是数据在系统各部件之间流动的情况，
、原型化方法主要用于解决什么问题？试说明这种方法的主要优缺点。11
经理和其他非技用户、原型化方法主要用于解决在产品开发的早期阶段需求不确定的问题，
这种方法 术项目风险承担者发现在确定和开发产品时，原型可以使他们的想象更具体化。
的主要优点如下：
原型作为一种需求工具，它初步实现所理解的系统的一部分，用户 明确并完善需求 ）1（
对原型的评价可以指出需求中的许多问题。
使系统可以探索不同的用户界面技术，原型作为一种设计工具， 探索设计选择方案 ）2（
达到最佳的可用性，并且可以评价可能的技术方案。
- 8 -

**原型作为一种构造工具，是产品最初子集的完整功能实现，通过 发展为最终的产品 ）3（
一系列小规模的开发循环，你可以完成整个产品的开发。
这种方法的主要缺点包括： ）4（
用户看到一个正在运行的原型便以为产品即将完成，他们会要求不再进行产品重建， ）1（
而只是对原型进行一些修改就交付。由于原型没有考虑软件的总体
"质量和可维护性，交付原型往往造成 的情况。"欲速则不达
也可能可能会采用不合适的操作系统或程序设计语言，开发人员为了快速构造原型， ）2（
于是便在他们往往已经习惯了这些选择，在一段时间的开发之后，使用一些效率低的算法。
系统中参杂了这些不理想的选择。
需求分析 第三章
、什么是需求分析？需求分析阶段的基本任务是什么？1
将用户非形式的需求陈进行细致的调查分析，开发人员准确地理解用户的要求，需求分析：
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：双方确定对问题的综合需求，这些需求包括功能需求，性能需求，环境需求，
用户界面需求。
。 ⑵分析与综合，导出软件的逻辑模型
修改完善软","确认测试计划","初步用户使用手册","需求规格说明书"⑶编写文档：包括编写
。" 件开发计划
、什么是结构分析方法？该方法使用什么描述工具？2
，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SA结构化分析：简称
中间层则是从抽象到底层具体地画出系统工程的每个细节。顶层抽象地描述整个系统，略。
具体的过渡。
判定树描述加工逻辑。判定表，使用结构化语言，作为描述工具，数据字典，使用数据流图，
、结构化分析方法通过哪些步骤来实现？3
⑶ ⑵抽象出当前系统的逻辑模型。 获得当前系统的物理模型。⑴了解当前系统的工作流程，
⑷作进一步补充和优化。 建立目标系统的逻辑模型。
4 图中包含哪几种种基本成分？E-R、
、关系和属性)即数据对象(实体
、联系可分为哪几种类型？5
∶(M，多对多联系N)∶(1，一对多联系1)∶(1一对一联系N)
、为何要进行数据结构规范化？6
为减少数据冗余，避免出现插入异常或删除异常，简化修改数据的过程。
、第一范式7
每个属性值都必须是原子值，即仅仅是一个简单值而不含内部结构。
、第二范式8
而不是由关键字的一部分(满足第一范式条件，而且每个非关键字属性都由整个关键字决定
，消除了非关键字属性对关键字属性的部分依赖。)来决定
、第三范式9
而且一个非关键字属性不能仅每个非关键字属性都仅由关键字决定，符合第二范式的条件，
即一个非关键字属性值不依赖于另一个非关键字(仅是对另一个非关键字属性的进一步描述
，消除了非关键字属性对关键字属性的传递依赖。)属性值
表SLOC表示系名，SDEPT表示学号，SNO，这里S-L-C(SNO,SDEPT,SLOC,CNO,G)、10**
- 9 -

**SLOC; SNO→G; SDEPT→: (SNO,CNO)函数依赖有 表示成绩。G表示课程号，CNO示楼号，
，对关系进行规范化，达到三范式要求。SDEPT→
SC(SNO,CNO,G)）1（S-D(SNO,SDEPT,) (3) S-L(SDEPT ,SLOC) ）2（ 和
、状态转换图11
来表示系统的通过描绘系统的状态及引起系统状态转换的事件，)简称为状态图(状态转换图
行为。提供了行为建模机制。
、层次方框图12
树形结构的顶层是一层次方框图用树形结构的一系列多层次的矩形框描绘数据的层次结构。
最底层下面的各层矩形框代表这个数据的子集，它代表完整的数据结构，个单独的矩形框，
。)不能再分割的元素(的各个框代表组成这个数据的实际数据元素
图IPO、13
图是输入、处理、输出图的简称，能够方便地描绘输入数据、对数据的处理和输出数据IPO
之间的关系。
、从哪些方面验证软件需求的正确性14
一致性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。(1)
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。(2)
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。对硬件技术(3)
只能从现有技术水平出发判断需对软件技术的进步则很难做出预测，的进步可以做些预测，
求的现实性。
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。(4)
总体设计 第四章
、什么是总体设计？总体设计的任务是什么？1
建立软件系统结基本任务是：确定系统功能是实现。总体设计又称为概要设计或初步设计。
、构（划分模块、定义模块功能、模块间的调用关系、定义模块的接口、评价模块的质量）
、编写概要设数据结构和数据库的设计（数据结构设计、概念设计、逻辑设计、物理设计）
。计文档（概要设计说明书、用户手册、数据库设计说明书、修订测试计划）
、总体设计分为哪两个阶段？2
系统设计阶段，确定系统的具体实现方案；结构设计阶段，确定软件结构。
、总体设计过程包括几个步骤？3
设想供选择的方案1.
选取合理的方案2.
推荐最佳方案3.
功能分解4.
设计软件结构5.
设计数据库6.
制定测试计划7.
书写文档8.
审查和复审9.
、模块化4
是把程序划分成独立命名且可独立访问的模块，把这些模块每个模块完成一个子功能，
集成起来构成一个整体，可以完成指定的功能满足用户的需求。
、模块化的优点5
）模块化使软件容易测试和调试，因而有助于提高软件的可靠性。1（** 
- 10 -

）模块化也有助于软件开发工程的组织管理。2（
、抽象6
就是抽出事物的本质特性而暂时不考虑它们的细节。
、抽象化包括哪些？7
、数据抽象。 过程的抽象
、模块的独立程度的定性标准度量是什么？好的模块应满足什么标准？8
内聚和耦合；高内聚低耦合。
、什么耦合？模块间的耦合有哪些？9
的紧密程度；有非直接耦合、标记耦合、数据耦)连接(指衡量不同模块彼此间互相依赖
。 、外部耦合、公共耦合和内容耦合 、控制耦合 合
、非直接耦合10
如果两个模块之间没有直接关系，它们之间的联系完全是通过主模块的控制和调用来实现
的，这就是非直接耦合。这种耦合的模块独立性最强。
、标记耦合11
这个记录是某一数据结构的子结构，就是标记耦合。如果一组模块通过参数表传递记录信息，
而不是简单变量。
、数据耦合12
公共数据不是控制参数、( 彼此之间是通过简单数据参数如果一个模块访问另一个模块时，
来交换输入、输出信息的，则称这种耦合为数据耦合。 ) 结构或外部变量
、控制耦合13
名字等控制信息，明显地控制选择另一模块的功能，就如果一个模块通过传送开关、标志、
是控制耦合。
、外部耦合14
而且不是通过参数表传递该全一组模块都访问同一全局简单变量而不是同一全局数据结构，
局变量的信息，则称之为外部耦合。
、公共耦合15
公共的数据环则它们之间的耦合就称为公共耦合。若一组模块都访问同一个公共数据环境，
境可以是全局数据结构、共享的通信区、内存的公共覆盖区等。
、内容耦合16
如果发生下列情形，两个模块之间就发生了内容耦合
; 一个模块直接访问另一个模块的内部数据(1)
; 一个模块不通过正常入口转到另一模块内部(2)
); 只可能出现在汇编语言中(两个模块有一部分程序代码重迭(3)
一个模块有多个入口。(4)
17 、内聚
， 通信内聚， 信息内聚， 功能内聚衡量一个模块内部各个元素彼此结合的紧密程度。
过程内聚，时间内聚，逻辑内聚，巧合内聚。
、功能内聚19
或者说该模块中所有部分一个模块中各个部分都是完成某一具体功能必不可少的组成部分，
都是为了完成一项具体功能而协同工作，则称该模块为功能内聚模不可分割的。紧密联系，
块。
、信息内聚20
每一项功能有一个唯一的入口各个功能都在同一数据结构上操作，这种模块完成多个功能，
由于这个模块的所有功能都是基确定该执行哪一个功能。这个模块将根据不同的要求，点。
- 11 -

，因此，它是一个信息内聚的模块。于同一个数据结构（符号表）
、通信内聚21
则称之为或产生了相同的输出数据，如果一个模块内各功能部分都使用了相同的输入数据，
通信内聚模块。通常，通信内聚模块是通过数据流图来定义的。
、过程内聚22
就得到过程内聚模把流程图中的某一部分划出组成模块，使用流程图做为工具设计程序时，
块。例如，把流程图中的循环部分、判定部分、计算部分分成三个模块，这三个模块都是过程内聚模块。
、时间内聚23
但模块的各个功能的执行与时间有这种模块大多为多功能模块，时间内聚又称为经典内聚。
关，通常要求所有功能必须在同一时间段内执行。例如初始化模块和终止模块。
、逻辑内聚24
这种模块把几种
数来确定该模块应执 相关的功能组合在一起，每次被调用时，由传送给模块的判定参
行哪一种功能。
、巧合内聚25
这种联系也很或者即使有联系，当模块内各部分之间没有联系，巧合内聚又称为偶然内聚。
松散，则称这种模块为巧合内聚模块，它是内聚程度最低的模块。
、扇出26
的模块数目。)调用(是一个模块直接控制
、扇入27
一个模块的扇入表明有多少个上级模块直接调用它。
、模块的作用域28
定义为受该模块内一个判定影响的所有模块的集合。
29 、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。
、模块化的优点。30
模块化使软件容易测试和调试，因而有助于提高软件的可靠性，提高软件的可修改 
性。
模块化也有助于软件开发工程的组织管理。 
、逐步求精、分层过程与抽象等概念之间的相互关系如何？31
即将软件的体系结构按自顶向提出的设计策略：Niklaus Wirth是”逐步求精自顶向下，“
直到用程序设计语言的语句能够实现对各个层次的过程细节和数据细节逐层细化，下方式，
为止，从而最后确立整个的体系结构。
采用抽象化的方在实施时，即分层的过程。这样的结构实际就是一个模块的分层结构，
可以使用问题所处环境的语言给出不同的抽象层次。在最高的抽象层次上，法，自顶向下，
在描述问题的解法则采用过程化的方法。而在较低的抽象层次上，概括地描述问题的解法。
应但最后在最低的抽象层次上，我们可以配合使用面向问题的术语和面向现实的术语。时，
使用能够直接实现的方式来描述这个解法。
、什么是软件结构图？简述软件结构设计优化准则。32
即将来程序的控制体系。软件系统的模块层次结构，反映了整个系统的功能实现，准则：高
模块大小适中、扇出适当、扇入、宽度、软件的深度、作用范围在控制范围内、内聚低耦合、
接口简单清晰易懂。
、什么是模块的控制范围和作用范围？他们的关系如何？33
- 12 -

模块中的一个判定影响的所有模作用范围：是模块本身和它的下属模块的集合。控制范围：
块的集合。作用范围应该在控制范围内。
、什么是变换流？什么是事物流？33
、输出三部分组成。变换流由输入、变换（或处理）
并根据输入选形成许多加工路径，事物流某个加工将它的输入流分离成许多发散的数据流，
择其中一个路径来执行。
"事物分析"﹑"变换分析"、试述34 设计步骤。
变换分析设计步骤：
⑴确定变换中心，逻辑输入和输出。
顶层的功能是完成所有模块的控制（名称应:变换结构--⑵设计软件结构的顶层和第一层
是系统名，以体现完成整个系统功能）
第一层至少有三种功能模块：输入﹑输出和变换模块
⑶设计中﹑下层模块，对第一层的模块自顶向下分解。
输入模块由两部分组成：接收数据﹑转换成调用模块所需信息。
输出模块由两部分组成：将数据转换成下属模块所需的形式﹑ 发送数据
变换模块：对每个基本加工建立一功能模块
：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。 ⑷设计的优化
其它输入模块与转换数据模块都很简单时，可将它们合并。
输出部分求精：每个物理输出设置专门模块，其它也可适当合并。
变换部分求精：根据设计准则，对模块适当合并。
事务分析的设计步骤：
⑴确定事务中心和加工路径
顶层模块有两个功能：接收数据和根据事务类型调 ⑵设计顶层（事务机构）和第一层
动相应处理模块。
⑶中下层模块的设计﹑优化工作与变换结构相同。
通常包括一调度模块，当事务类 发送分支出 接收分支 事务型软件结构包括两部分：
。 型不多时，可与主模块合并
转换为软件结构图。DFD、请将下图的35
解：
- 13 -

**详细设计 第五章** **详细设计的基本任务是什么？有那几种描述方法？ 、**1**设计模块的数据结构、设计数据库的物理结构、设计模块的详细算法、其它（代码设计、输
输出格式设计、人机对话设计）、编写详细设计说明书、评审。/入**
。PDL图、PAD程序流程图、
**语言描述下列程序的算法。**PDL**图和**PAD**、请使用流程图、**2
**中求最大数和次大数。**A(10)**～**A(1)**在数据**(1)
- 14 -

语言描述：PDL
GET(a[1],a[2],...a[10])
max=a[1];
max2=a[2];
FOR i=2 TO 10
IF a[i]>max
max2=max;
max=a[i];
ELSE
IF a[i]>max2
max2=a[i];
ENDIF
ENDIF
ENDFOR
PUT(max,max2)
END
、输入三个正整数作为边长，判断该三条边构成的三角形是等边、等腰还是一般三角形。3
语言描述。PDL图和PAD使用流程图、
解：如图：
- 15 -

语言描述如下：PDL
GET(a,b,c)
IF (a+b>c and b+c>a and c+a>b)
IF (a=b)
IF (a=c)
") 等边三角形PUT("
ELSE
等腰三角形PUT(" ")
ENDIF
ELSE
IF(a=c)
") 等腰三角形PUT("
ELSE
IF (b=c)
- 16 -

") **等腰三角形**PUT("
ELSE
") **一般三角形**PUT("
ENDIF
ENDIF
**ENDIF** 
ELSE
") **这不是一个三角形**PUT ("
ENDIF
**图。**PAD**写出的程序的**PDL**、画出下面用**4
WHILE P DO
IF A >O THEN A1 ELSE A2 ENDIF;
S1; **IF B>0 THEN B1;** 
WHILE C DO S2;S3 ENDWHILE;
ELSE B2
ENDIF;
B3
ENDWHILE;
**编码与测试 第六章**
1? **软件测试中，应注意哪些原则**?**、软件测试的目的是什么** **答：软件测试的目的是为了发现软件的错误。** **软件测试中应注意的原则有：
于对照检查，做到有的放** **测试用例应由输入数据和预期的输出数据两部分组成。这样便**(1)
**矢。
测试且例不仅选用合理输入数据，还要选择不合理的输入数据。这样能更多地发现错误，**(2)
**提高程序的可靠性。对于不合理的输入数据，程序应拒绝接受，并给出相应的提示。** **除了检查程序是否做了它应该做的事，还应该检查程序是否做了它不应该做的事。**(3)
**应制定测试计划并严格执行，排除随意性。**(4)
**长期保留测试用例。**(5)
- 17 -

对发现错误较多的程序段，应进行更深入的测试。(6)
心理状态是测试自己程序的的行为，"挑剔性"测试是一种程序员应避免测试自己的程序。(7)
障碍。
试对他们的检错能力进行比较?有哪些覆盖标准?、什么是白盒测试法2?
测试人员须了解程序内部结构和处理过白盒法测试法把测试对象看作一个打开的盒子，答：
检验内部控制对程序中尽可能多的逻辑路径进行测试，以检查处理过程的细节为基础，程，
结构和数据结构是否有错，实际的运行状态与预期的状态是否一致。
白盒法有下列几种覆盖标准：
语句覆盖：设计若干个测试用例，运行被测程序，使得每一可执行语句至少执行一次。 
运行被测程序，设计若干个测试用例，判定覆盖： 使得程序中每个判断的取真分支和取
假分支至少经历一次。
使得程序中每个判断的每个条件的可运行被测程序，设计若干个测试用例，条件覆盖： 
能取值至少执行一次。
条件覆盖：设计足够的测试用例，使得判断中每个条件的所有可能取值至少执行/判定 
一次，同时每个判断中的每个条件的可能取值至少执行一次。
使得每个判断的所有可能的条件运行被测程序，设计足够的测试用例，条件组合覆盖：
取值组合至少执行一次。
路径覆盖：设计足够的测试用例，覆盖程序中所有可能的路径。 
凡满足其其中条件组合发现错误的能力较强，从上到下的覆盖标准其检错能力也从弱到强，
一般以条件组合覆盖在实际的逻辑测试中，也必然满足前四种覆盖标准。标准的测试用例，
为主设计测试用例，然后再补充部分用例来达到路径覆盖的测试标准。
? 这些方法各有什么特点?采用黑盒技术测试用例有哪几种方法?、什么是黑盒测试法3
测试人员完全不考虑程序的内部结构和处黑盒测试法把被测试对象看成是一个黑盒子，答：
理过程，只在软件接口处进行测试，依据需求规格说明书，检查程序是否满足功能要求。
采用黑盒技术测试用例的方法有：
等价类的划分、边界值分析、错误推测和因果图。
(是将输入数据按有效的或无效的等价类的划分： 划分成若干个)也称合理的或不合理的
这样就把漫无边迹的随测试每个等价类的代表值就等于对该类其他值的测试。等价类，
用少量有代表性的例子代替大量测试目的相同的例机测试改为有针对性的等价类测试，
能够发现但这个方法的缺点是没有注意选择某些高效的、能有效地提高测试效率。子，
更多错误的测试用例。
刚刚大于和刚刚小于选取正好等于、是将测试边界情况作为重点目标，边界值分析法： 
边界值的测试数据。这种方法)边界情况是指输入等价类和输入等价类边界上的情况。(
可以查出更多的错误，因为在程序中往往在处理边界情况时易发生错误。
从而人们根据经验或直觉推测程序中可能存在的错误，是在测试程序时，错误推测法： 
有针对性地编写检查这些错误的测试用例。
它的基本原理是通过能够有效地检测输入条件的各种组合可能会引起的错误。因果图： 
把用自然语言描述的功能说明转换为判定表，画因果图，最后为判定表的每一列设计一
个测试用例。
这几种方法都不能提供一组完整的测试用例，在实际测试中应把各种方法结合起来使用。
综合策略：就是联合使用上述几种测试方法，尽可能多地发现程序中的错误。
? 这些测试与软件开发各阶段之间有什么关系?、软件测试要经过哪些步骤4
- 18 -

答：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。
检查各个模块是否正确实现规定的功对源程序中每一个程序单元进行测试，单元测试： 
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。
也就是检查概要设计是是为了检查与设计相关的软件体系结构的有关问题，集成测试： 
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。
)如硬件、其他支持软件、数据、人工等(系统测试：是把已确认的软件与其他系统元素 
结合在一起进行测试。以确定软件是否可以支付使用。
? 测试中采用什么方法?、单元测试有哪些内容5
答：单元测试主要针对模块的以下五个基本特征进行测试：
模块接口(1)
局部数据结构(2)
重要的执行路径(3)
错误处理(4)
边界条件(5)
通过驱动模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。
接收以上级模块调用被测模块的格式驱动被测模块，块来模拟被测试模块的上级调用模块，
它的作用是返回桩模块则用来代替被测试模块所调用的模块。被测模块的测试结构并输出。
被测模块所需的信息。
6? 渐增式测试如何组装模块? 非渐增式测试与渐增式测试有什么区别?、什么是集成测试
将所有模块按照设计要求组装成一个完整的系统进集成测试是指在单元测试的基础上，答：
行的测试。
再把所有模块组装成一个完整的系统非渐增式测试是指首先对每个模块分别进行单元测试，
而渐增式测试就是逐个把未经测试的模块组装到已经过测试的模块上去进行集进行的测试。
每加入一个新模块进行一次集成测试，成测试，渐增式测试重复此过程直到程序组装完毕。
有两种不同的组装方法：自顶向下和自底向上结合。
两者区别是：
非渐增式方法把单元测试和集成测试分成两个不同的阶段，前一阶段完成模块的单元测(1)
同时完成。而渐增式测试往往把单元测试和集成测试合在一起，后一阶段完成集成测试。试，
非渐增式需要更多的工作量，因为每个模块都需要驱动模块和桩模块，而渐增式利用已(2)
测试过的模块作为驱动模块或桩模块，因此工作量少。
渐增式可以较早地发现接口之间的错误，非渐增式最后组装时才发现。(3)
渐增式有利于排错，发生错误往往和最近新加入的模块有关，而非渐增式发现接口错误(4)
推迟到最后，很难判断是哪一部分接口出错。
渐增式比较彻底，已测试的模块和新的模块再测试。(5)
渐增式点用时间较多，但非渐增式所需更多的驱动模块和桩模块也占用一些时间。(6)
非渐增式开始可并行测试所有模块，能充分利用人力，对测试大型软件很有意义。(7)
? 该阶段有哪些工作?、什么是确认测试7
它的任务是检查软件的功能与性能是否与需求规格说明书中确认测试又称有效性测试。答：
进行确认确认测试阶段有两项工作：因而需求说明是确认测试的基础。确定的指标相符合。
测试与软件配置审查。
- 19 -

? 调试有哪些技术手段?、调试的目的是什么8
调试的目的是确定错误的原因和位置，调试则是在进行了成功的测试之后才开始的工作。答：
。调试的技术手段有简单的调试方法、归纳法、(Debug)并改正错误，因此调试也称为纠错
演绎法和回溯法等。
、将正确答案的编号填入题目空白处：9
在白盒测试用例中，有语句覆盖、条件覆盖、判定覆盖、路径覆盖等，其中是最强的覆(A)
是两个Y，X盖准则。为了对下图所示的程序进行覆盖测试，必须适当地选取测试数据。若
如表中给出(四组IV，III，II，I变量，可供选择的测试数据组共有，则实现语句覆盖至少)
；实现路径覆盖至(C)；实现条件覆盖至少应采用的测试数据组是(B)应采用的测试数据组是
。(E)或(D)少应采用的测试数据组是
可供选择的答案：
A 路径覆盖(4) 判定覆盖(3) 条件覆盖(2) 语句覆盖(1)：
(1)I：E～B和III，(6)II 组III和II、(5)I 组IV和(4)I 组IV和(3)III 级III和(2)II 组II和
组IV
组IV和II、(8)I 组IV和III，(7)I
(8) ：(5) E：(4) B:(1) C:(4) D：A答：
、请对“判定三角形类别”程序算法用等价类划分和边界值分析法设计测试用例，并检查10
逻辑覆盖标准。
解：先划分等价类并编号：
- 20 -

为合理等价类设计测试用例，对于表中对应的四个合理等价类，用三个测试用例覆盖。
为每一个不合理等价类设计一个测试用例：
用边界值法设计测试用例：
、某“调整工资”处理模块接受一个“职称”的变量，根据职称的不同（助教，讲师，副11
只有工龄超过两年才能调整工教授）作不同的处理，其中若是助教还必须输入工龄，教授，
资。请用等价类划分法设计测。
划分等价类：1.
计测试用例：
- 21 -

**软件维护 第七章** **、软件维护有哪些内容？**1**一些隐含的错误在某些特定的使用环境下会暴露在软件交付使用后，）校正性维护。**1**（答：
出来。这个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，** **过程就称为校正性维护。
）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**2**（** **化而修改软件的过程称为适应性维护。
）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**3**（** **称为完善性维护。** **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。预防性维护。）**4**（** **软件维护的特点是什么？** **、**2
**答：主要体现在三个方面：
若不采用软件的开发过程对软件的维护有很大的影响。非结构化维护和结构化维护。** **）**1**（
这是一种非结构维护工作非常困难，则软件只有程序而无文档，软件工程的方法开发软件，
容易进行维护工作，则各阶段都有相应的文档，若采用软件工程的方法开发软件，化的维护。** **这是一种结构化的维护。** **软件生存软件维护的困难性是由于软件需求分析和开发方法的缺陷。维护的困难性。** **）**2**（** **）**3**（周期中的开发阶段没有严格而又科学的管理和规划，就会引起软件运行时的维护困难。
这是软件维护有形的软件维护的费用在总费用中的比重是在不断增加的，软件维护的费用。
软件维护费用增加的占有原因是软件维即要占有更多的资源。另外还有无形的代价，代价。** **护的生产率非常低。** **软件维护的流程是什么？** **、**3**运行维护并做详细记答：软件维护的流程如下：提交维护申请报告、审查申请报告并批准、
录、复审。** **软件维护的副作用有哪些？** **、**4
**答：维护的副作用有以下三种：
编码副作用。在使用程序设计语言修改源代码时可能引入的错误。** **）**1**（** **因而导致有可能造成软件设计与数据结构不匹配，在修改数据结构时，数据副作用。** **）**2**（**
- 22 -

但它可以通过详细的设计文档加以数据副作用是修改软件信息结构导致的结果。软件错误。
控制。
文档副作用。如果对可执行软件的修改没有反映在文档中，就会产生文档副作用。 ）3（
、什么是软件可维护性？可维护性度量的特性是什么？5
答：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。软件的可
维护性可用以下七个质量特性来衡量，即可理解性、可测试性、可修改性、可靠性、可移植
性、可使用性和效率。
提高可维护性的方法有哪些？ 、6
）建立明确的软件质量目标。1（答：
）使用先进的软件开发技术和工具。2（
）建立明确的质量保证。3（
）选择可维护性的程序设计语言。4（
）改进程序的文档。5（
- 23 -