第一章

**一、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，他是包括程序、数据及其相关文档的完整集合。其中，程序是为实现设计的功能和性能的要求而编写的指令序列；数据是使指令能够正常操纵信息的数据结构；文档是与程序开发、维护和使用有关的图文资料。

**二、软件工程的主要原则包括那些方面？它的基本要素有哪几项？**

主要原则（七点）：

（1）用分阶段的生命周期计划严格管理软件工程过程。

（2）坚持在软件工程过程中进行阶段评审。

（3）实行严格的产品控制。

（4）采用现代的开发技术进行软件的设计与开发。

（5）工作结果应当是能够清楚地审查的。

（6）开发小组的人员应该是“少而精”。

（7）承认不断改进软件工程实践的必要性。

基本要素：方法、工具和过程。

软件工程方法为软件开发提供了“如何做某项工作”的技术指南。软件工程工具为软件工程方法提供了自动的或半自动的软件支撑环境。软件工程中的“过程”是将软件工程的方法和工具综合起来以达到合理、及时地进行计算机软件开发的目标。

**三、一个良好的软件工程应该具备哪些特点？**

（1）易理解性（2）可见性（3）可支持性（4）可接受性

（5）可靠性（6）健壮性（7）可维护性（8）高效率

**四、几种软件过程模型？**

（四种）1.线性顺序模型 2.原型模型 3.快速应用开发模型(RAD) 4.演化软件过程模型（其又分为：增量模型，螺旋模型）。

第二章

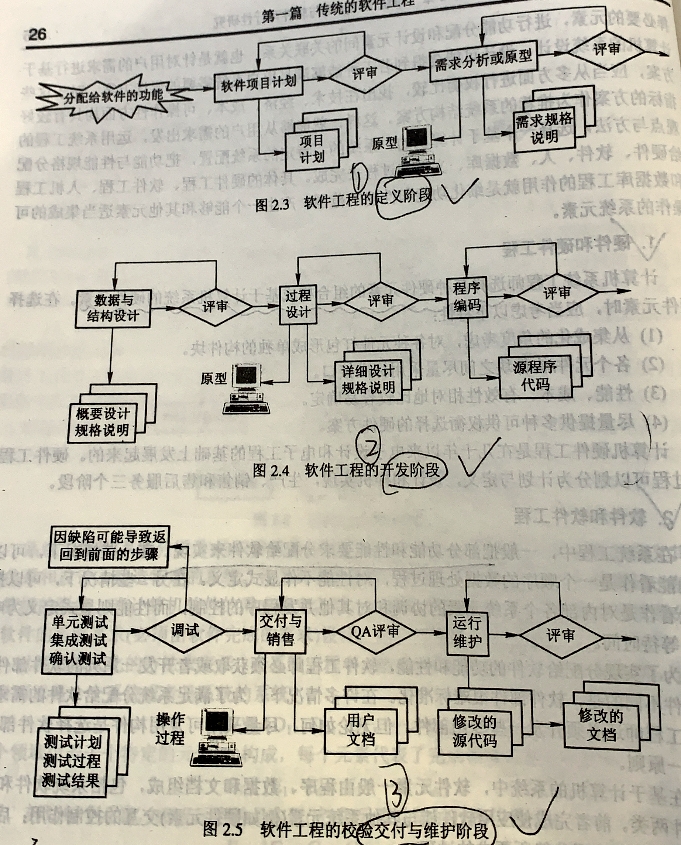
**一、软件工程活动可以大略划分为哪几个阶段？请简单介绍各个阶段的主要任务。**

软件工程分为三个阶段，即**定义、开发、检验交付与维护**阶段。

定义阶段：开发人员通过分配给软件的功能，来制定相应地软件项目计划，再对这个计划进行评审，不断地进行改进，最后对其需求分析或原型进行不断的评审和改进。

开发阶段：开发人员结合数据与结构设计生成概要设计规格说明，再对其进行不断的评审改进；在过程设计的同时会生成详细的规格说明，再进行评审改进；最后进行程序编码，形成源程序代码，不断进行评审改进。

检验交付与维护阶段：对源程序代码进行不断的调试，在通过QA评审后，进行交付与销售；最后进入运行维护，写给错误代码和生成新的修改后的文档。



第三章

**一、什么是需求分析？**

在可行性研究阶段的成果上，进一步的将用户的需求具体化，全面的理解和恰当的表达需求。

**二、需求分析的任务是什么？**

主要任务是通过软件开发人员与用户的交流和讨论，准确的获取用户对系统的具体要求。

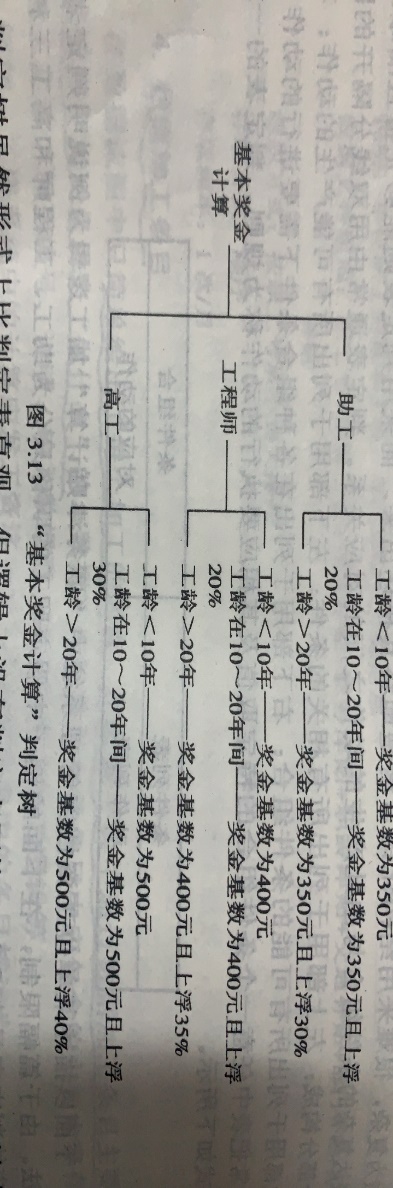
**三、需求分析的步骤？**

1. 通过调查研究，获取用户的需求
2. 去除非本质因素，确定系统的真正需求
3. 描述需求，建立系统的逻辑模型
4. 书写需求说明书，进行需求复审

**四、需求分析原则是什么？**

1. 能够表达和理解问题的数据域和功能域
2. 能够将复杂问题分解化简
3. 能够给出系统的逻辑表示和物理表示

**五、判定树**



第四章

**一、什么是模块化？模块化的好处和特点是什么？**

模块：具有相对独立性的，由数据说明、执行语句等程序对象构成的集合。

模块化：将整个程序划分为若干个模块，每个模块用于实现一些特定的功能。

好处：模块化可以降低解决问题的复杂度，从而降低软件开发的难度，而且可以使程序结构清晰，增加易读性和可修改性；还有利于提高代码的可重用性及团队合作开发大型软件的可行性。

特点：一个模块具有输入/输出（接口）、功能、内部数据和程序代码四个特征。输入输出用于实现模块与其他模块间的数据传送，功能指模块所能完成的工作，内部数据指仅能在模块内部使用的局部量，程序代码用于描述实现模块功能的具体方法和步骤。

**二、什么是模块化的独立性？**

衡量模块化的定性指标有两个：耦合性、内聚性。

1. 耦合性（模块之间的东西）

耦合性是对一个软件结构内部不同模块间联系紧密程度的度量指标。模块间的联系越紧密，耦合性就越高，模块的独立性也就越低。

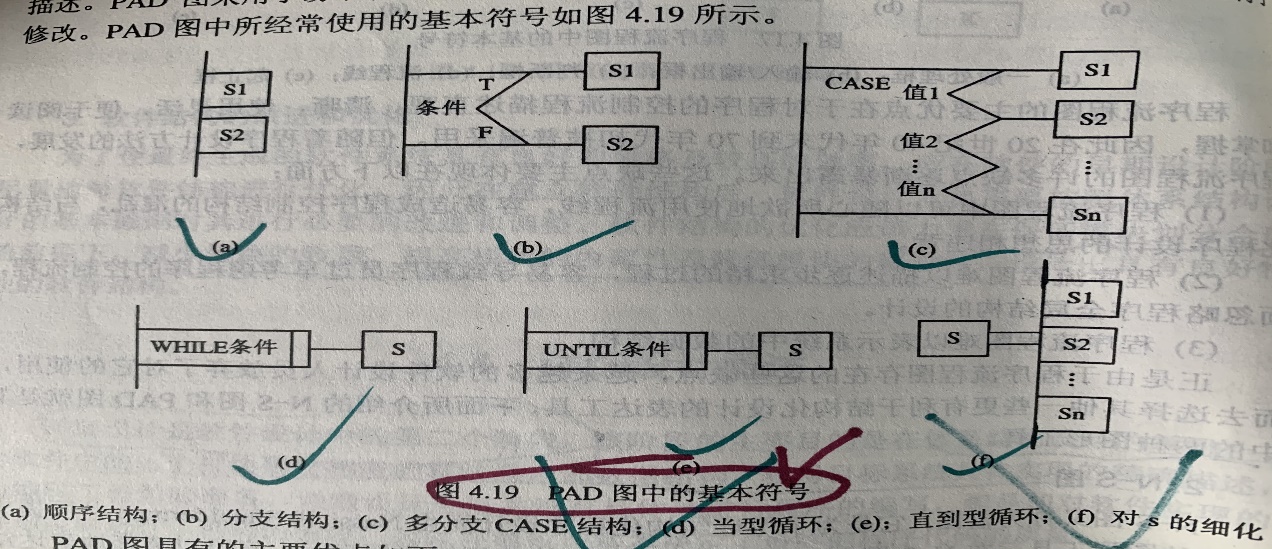
1. 数据耦合：两个模块间仅通过参数交换信息，且交换的信息全部为简单数据。
2. 公共耦合：两个或多个模块通过引用公共数据相互联系。
3. 控制耦合：模块之间交换的信息中包含有控制信息。
4. 内容耦合：一个模块对另一个模块中的内容进行了直接的引用甚至修改，或通过非正常的入口进入到另一模块的内部，或一个模块具有多个入口，或两个模块共享一部分代码。
5. 内聚性（模块内部的东西）

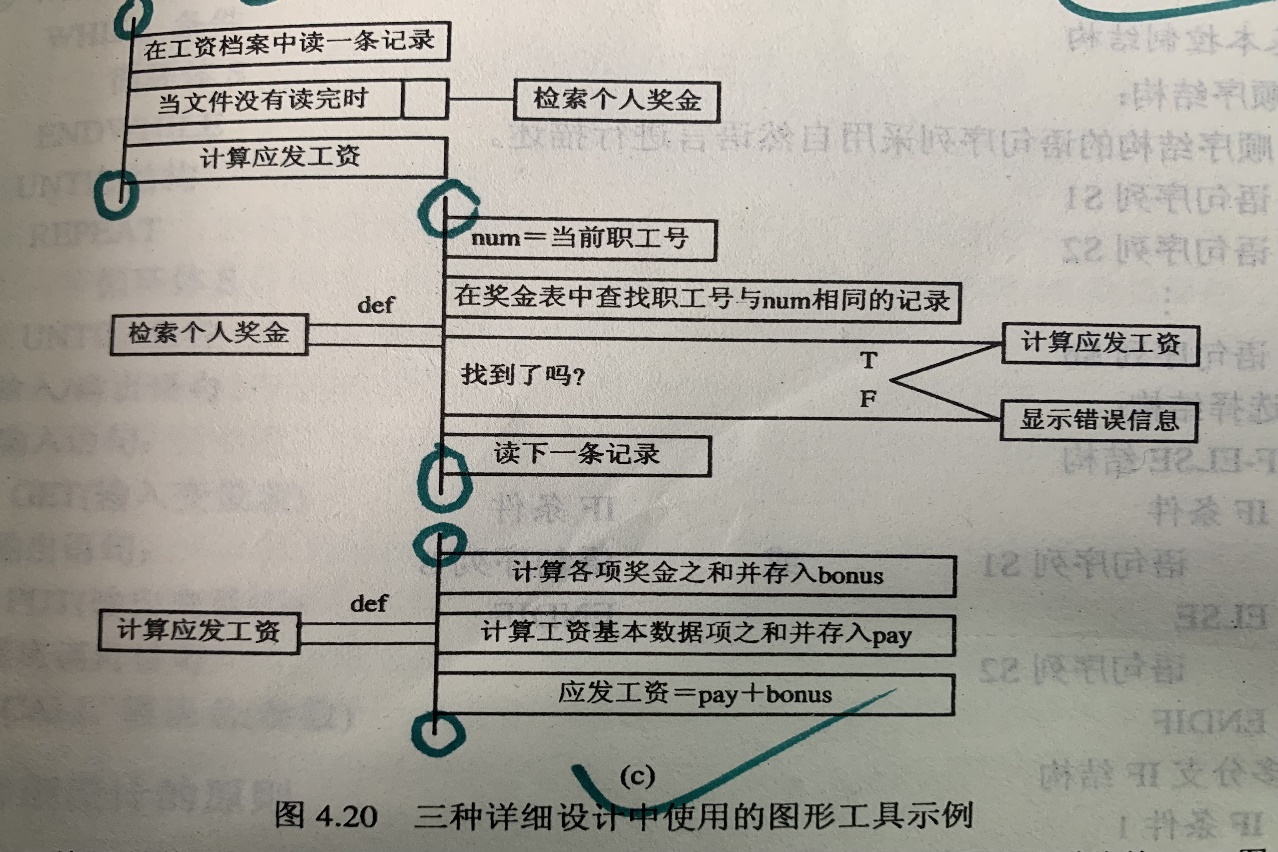
内聚性是对一个模块内部各个组成元素之间相互结合的紧密程度的度量指标。模块中组成元素结合的越紧密，模块的内聚性就越高，模块的独立性也就越高。

（1）偶然内聚 （2）逻辑内聚 （3）时间内聚 （4）过程内聚

（5）通信内聚 （6）顺序内聚 （7）功能内聚

**三、PAD图**





第六章

**一、软件工程的概念、目的是什么？**

概念：

软件测试是在软件投入运行前对软件需求分析、软件设计规格说明和软件编码进行查错和纠错。查错的活动称测试，纠错的活动称调试。可以说软件测试是为了发现错误而执行程序的过程。或者说，软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例，并利用这些测试用例去运行程序，以发现程序错误的过程。

目的：

1. 测试是一个为了寻找错误而运行程序的过程；
2. 一个好的测试用例是指很可能找到迄今为止尚未发现的错误的用例；
3. 一个成功的测试是指揭示了迄今为止尚未发现的错误的测试。

**二、软件测试的方法（技术）？**

软件测试技术分为：静态分析技术、动态分析技术。

1. 静态分析技术

静态分析技术**不执行被测试软件**，可对需求分析说明书、软件设计说明书、源程序作结构检查、流图分析、符号执行等来找出软件错误。

1. 动态分析技术

动态分析时**执行被测程序**，由执行结果分析程序可能出现的错误。

动态测试包括：功能测试（黑盒测试）、结构测试（白盒测试）。

1. 白盒测试 —— 测试逻辑结果

白盒测试又称为结构测试或逻辑驱动测试，此方法是将测试对象比作一个打开的盒子，他允许测试人员利用程序内部的逻辑结构和相关信息来设计或选择测试用例，对穿过软件的逻辑路径进行测试，可以在不同点检查程序的状态，以确定实际状态与预期状态是否一致。

使用白盒方法测试程序模块的检查点主要包括：对程序模块的所有独立的执行路径应至少测试一次；对所有的逻辑判定，取“真”与取“假”两种情况都能至少测试一次；在循环的边界和运行界限内执行循环体；测试内部数据结构的有效性等。

白盒测试的覆盖技术（8种）：

a.语句覆盖（最弱） b.判定覆盖 c.条件覆盖 d.判定/条件覆盖 e.条件组合覆盖 f.点覆盖 g.边覆盖 h.路径覆盖

1. 黑盒测试 —— 测试功能结果

黑盒测试方法（又称功能测试）是在已知产品应该具有的功能的情况下，通过测试来检验是否每个功能都能正常使用的测试方法。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。黑盒测试是在程序接口进行的测试，他只检查程序功能是否能按照规格说明书的规定正常使用，程序是否能适当的接收输入数据产生正确的输出信息，并且保持外部信息的完整性。

黑盒测试的用例设计：

1. 等价类及其划分（过程）
2. 划分等价类：典型的、实用的重要测试方法。把程序得到输入数据集合按输入条件划分为若干等价类。（分为有效等价类和无效等价类）
3. 确定测试用例。
4. 用等价类划分法设计测试用例的案例。
5. 边界值分析
6. 边界值分析。
7. 确定测试用例。
8. 因果图法
9. 设计测试用例。
10. 利用因果图设计测试用例的实例
11. 错误推测法

无固定步骤，主要依靠经验的不断累积

**三、软件测试策略**

1.单元测试：在于发现各模块内部可能存在的各种差错

单元测试内容（五个）：

1. 测试模块接口
2. 测试局部数据结构
3. 重要的执行路径
4. 错误处理
5. 边界测试

单元测试步骤：

（1）设计辅助测试模块

（2）编写测试数据

（3）进行多个单元的并行测试

2.集成测试

也称综合测试，组装测试，联合测试。是按设计要求把通过单元测试的各个模块组装在一起之后进行测试，便于发现与接口有关的各种错误。

（1）非渐增式测试

先测试每个模块，再把所有模块按设计要求放在一起，组合成所要的程序再进行测试

（2）渐增式测试

把下一个要测试的模块同已经测试好的那些模块结合起来进行测试，测试完以后再把下下一个应该测试的模块结合起来测试，这种测试每次增加一个模块。

分成两种方式：①自顶向下结合②自底向上结合

1. 确认测试（α、β测试）

也称有效性测试、合格性测试、验收测试。主要由使用用户参与测试，检验软件规格说明的技术标准的符合程度，是保证软件质量的最后关键环节。

步骤：①确认测试标准；②配置复审。

**α、β测试：**

α测试是由一个用户在开发环境下进行测试，也可以是开发机构的内部人员在模拟实际操作环境下进行的测试。（用户+开发人员）

β测试是由软件的多个用户在一个或者多个用户的实际使用环境下进行的测试。（生产+用户）

1. 系统测试

将通过确认测试的软件，作为整个基于计算机系统的一个元素，与计算机硬件，外设，某些支持软件，数据和人员等其他系统元素结合在一起，在实际运行使用环境中，对计算机系统进行一系列的组装，测试和确认测试。

1. 恢复测试：主要检查系统的容错能力。
2. 安全性测试：要检验在系统中已存在的系统安全性措施、保密性措施是否发挥作用，有无漏洞。
3. 强度测试：检查在系统运行环境不正常到发生故障的时间内，系统可以运行到何种程度的测试。
4. 性能测试：测试软件在被组装进系统的环境下运行的性能。

第七章

**一、什么是面向对象？**

面向对象是相对于面向过程的一种编程方式。面向对象的编程方式，操作的是一个个的对象，面向对象（oop）：将需要做的事，分为多个模块（对象），侧重点在于受影响的对象。

**二、什么是类？**

对象是指现实世界或者概念世界中的任何事物，类是具有相同结构特征的对象的结构抽象。此结构特征包括对象的属性特征和操作接口特征。

**三、什么是对象及对象实例？**

现实世界中的具体事物就是对象或者对象实例，类则是对象实例的结构抽象。

**四、什么是消息？**

消息就是现实世界中的请求或者通知事件，这一般都是通过对系统执行一定的操作完成的。在面向对象程序中，调用一个对象的方法或者操作叫做向该对象发消息，调用者叫做消息发送者，被调用者叫做消息接收者。

**五、面向对象技术的基本特点（封装性、继承性、多态性、抽象性）**

封装性：实现的是信息隐藏。仅将需要向外公开的方法和属性向外公开；所有不需要向外公开的方法和属性都被隐藏起来了。

继承性：一种高级的代码重用技术，子类自动继承父类的所有代码并且可以进行任意的覆盖和扩充。

多态性：同一事物可以有多种不同的形态或含义，也可以认为是从不同的角度观察同一事物，可以得到不同的视图。

抽象性：对复杂事物本质和特性的提炼和概括的能力，可以说没有抽象就没有理论思维，就没有指导认识世界改造世界的一般性结论，也就没有系统分析和设计。

**六、UML图（分为五类图形）**

1. 用例图
2. 静态图：类图、对象图和包图
3. 行为图：状态图、活动图
4. 交互图：顺序图、协作图
5. 实现图：构件图、配置图

第十五章

1. **项目管理的任务是什么？（6点）**
   1. 启动一个软件项目：包括明确目标、界定范围、初步确定解决方案。
   2. 项目度量：包括对过程的度量和对产品的度量是两个方面。对过程的度量（如进度，成本等等）是为了改进开发过程，对产品的度量是为了保证产品质量。概括的说，度量是为了产品和过程进行定量的管理。
   3. 估算：包括对项目规模、工作量、成本、进度、资源等各个方面的估算。估算的目的是制定科学的计划。估算的依据是经验公式和历史数据。
   4. 风险分析：针对存在的“不确定性”可能导致的问题，事先进行分析，以便最大限度地规避开发过程中可能发生的风险。风险分析实际上就是贯穿在软件工程过程中的一系列风险管理步骤。包括风险识别，风险预测、风险跟踪等保护性活动。
   5. 制定计划：项目管理者要使工作能够高效率地、有条不紊地进行，就必须指定开发计划。开发计划应当界定每项细分地工作任务所需要地时间、人力资源、阶段划分、工作产品形式、启动/结束条件等。
   6. 实施跟踪和控制：项目管理中，必须以计划为准绳，以度量数据为依据，对实际工作进行跟踪和检查，出现较大偏差时要及时进行调控。实现对软件开发过程地有效控制。
2. **项目管理的问题是什么？**

所谓问题，包括需求问题和工程过程问题两重含义。

问题管理要解决两个问题：问题界定与问题的分解划分。

问题界定要明确就当前认识层次而言，确定的软件范围是什么。

软件范围的确定可以从三个层面描述：

(1) 工程背景与约束条件(环境需求)

(2) 信息目标(数据I/O需求)

(3) 功能和性能(综合需求)

问题分解又称为划分，首先对功能进行细分从整体功能到子系统，从子系统到功能模块。其次，要对生产待交付产品的工作过程进行细分。

第十六章

**1.软件度量的目的？**

目的：评价软件产品的质量，显化软件开发的生产率，给出使用了新的软件工程方法和工具所产生的生产率方面和质量方面的效率，建立项目估算的基线，帮助调整对新工具和培训的要求。

度量的方式分为：直接度量和间接度量

软件度量的三个方面：

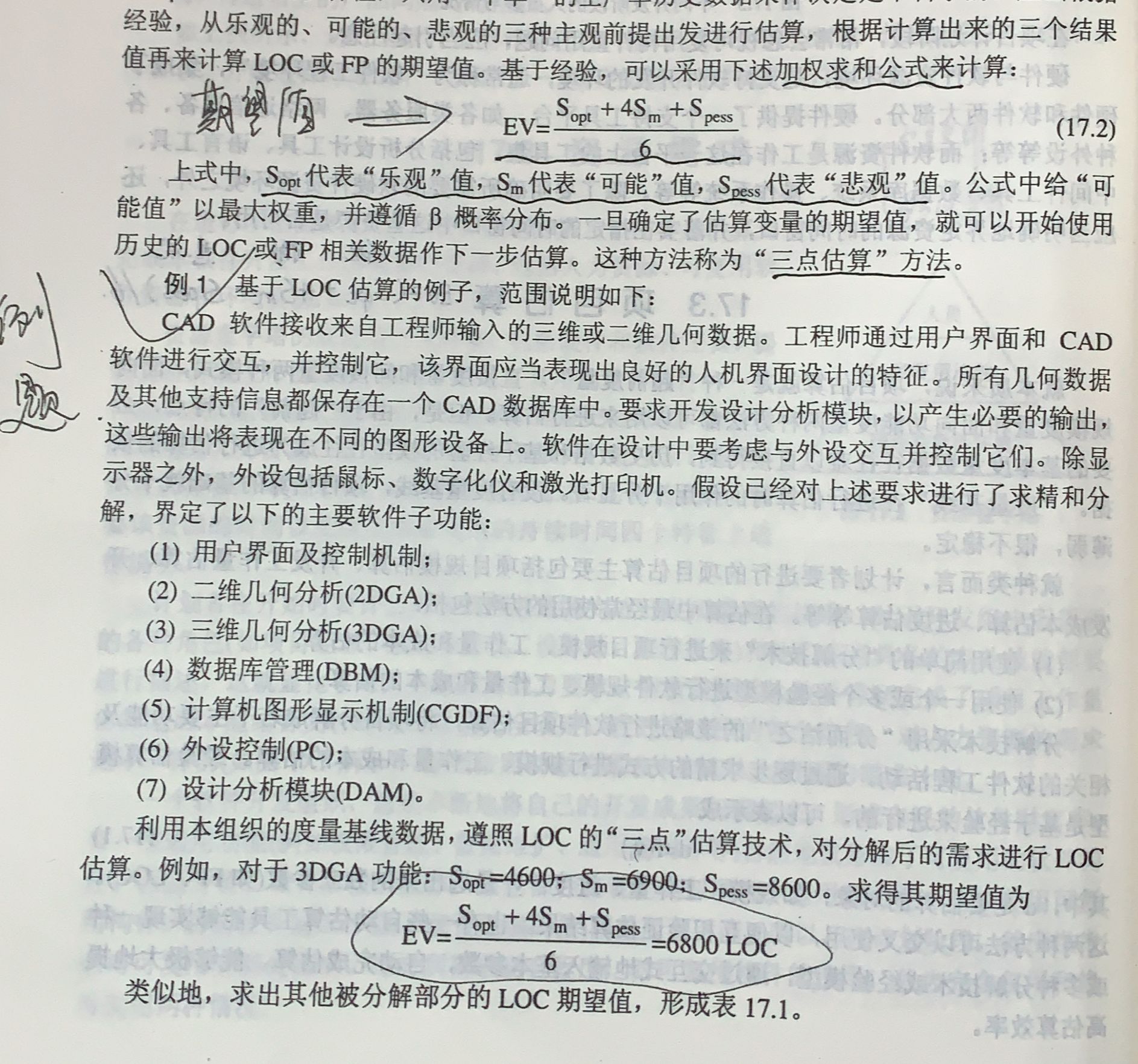
（1）面向规模的度量：是对软件产品和软件开发过程的直接度量。是通过规范化质量和生产率的测量得到的。

（2）面向功能的度量：是对软件和软件开发过程的一种间接度量方法。把注意力放在未来软件应当满足的功能性、实用性作为度量的原始依据。

（3）软件质量的度量：是一种保护性活动，贯穿软件工程过程的始终。

第十七章

**计算期望值**



第十八章

1. **什么是软件工程风险管理？**

风险管理是对项目风险进行识别、分析、应对和监控的过程，是项目管理中很重要的管理活动，有效的实施软件风险管理是软件项目开发工作顺利完成的保证。

风险管理的达成必须包括三个要素：

1.在项目开发计划中必须制定风险管理计划；

2.在项目预算中必须包含解决风险所需的经费；

3.评估风险时，风险的影响也必须纳入项目计划中。

1. **软件风险所包含的内容、活动？**

软件风险管理所包含的内容： 软件项目风险是指在软件开发过程中遇到的预算和进度等方面的问题以及这些问题对软件项目的影响。软件项目风险会影响[项目计划](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E8%AE%A1%E5%88%92/9817906)的实现，如果项目风险变成现实，就有可能影响项目的进度，增加项目的成本，甚至使软件项目不能实现。

风险管理实际上是四个不同的活动：风险识别、风险评估和风险缓解与监控。

第十九章

1. **什么是软件质量？**

软件质量：与软件产品满足规定和隐含需求的能力有关的全体特征（或特性）。

1. **什么是SQA？什么是SQA所包含的活动？**

软件质量保证（SQA）是一种应用于整个软件工程的保护性活动。

SQA小组的职责是辅助软件工程小组得到高质量的最终产品。

具体的SQA活动应当包括：

1. 为项目准备SQA计划
2. 参与开发该项目的软件过程
3. 复审各项软件工程活动；对工程活动是否符合定义好的团建工程过程进行核实。
4. 审查指定的软件工作产品
5. 确保软件工作及工作产品中的偏差已记录在案
6. 记录所有的不符合部分，并报告给高级管理者，对不符合部分进行跟踪，直到问题得到解决。
7. SQA小组还要协调变更的控制和管理，并协助收集项目度量信息。

第二十章

1. **什么是SCM？**

软件配置管理（SCM）是贯穿于整个软件工程过程活动中的一种保护性活动。因为在软件工程中，变化随时可能发生（需求、设计、编码、文档等等均可能变化），所以设计了SCM活动来标识变化、控制变化，保证变化被适当的实现以及向其他人员报告变化。它的主要目的是使变化可以更容易地被适应，并减少当变化必须发生时所需花费的工作量。