

1 软件危机的主要表现是什么？

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。

软件危机的主要表现：

1. 对软件开发成本和进度的估计不准确
2. 用户不满意
3. 软件质量不高、可靠性差
4. 软件常常不可维护、错误难以改正。
5. 缺乏适当的文档资料
6. 软件成本占系统总成本的比例逐年上升
7. 软件开发速度跟不上计算机发展速度

产生软件危机的原因：

1. 与软件本身的特点有关

软件不同于硬件，它是计算机系统的逻辑部件而不是物理部件。在写出程序代码并在计算机运行之前，软件开发过程的进展情况较难衡量，软件开发的质量也较难评价。因此，管理和控制软件开发过程相当困难。

2. 软件不易于维护

(1)软件维护通常意味着改正或修改原来的设计，客观上使软件较难维护。

(2)软件不同于一般程序，它的规模大，不易于维护。

3.在软件开发过程中，或多或少地采用了错误的方法和技术。

4.对用户需求没有完整准确的认识，就匆忙着手编写程序。

解决软件危机的途径：

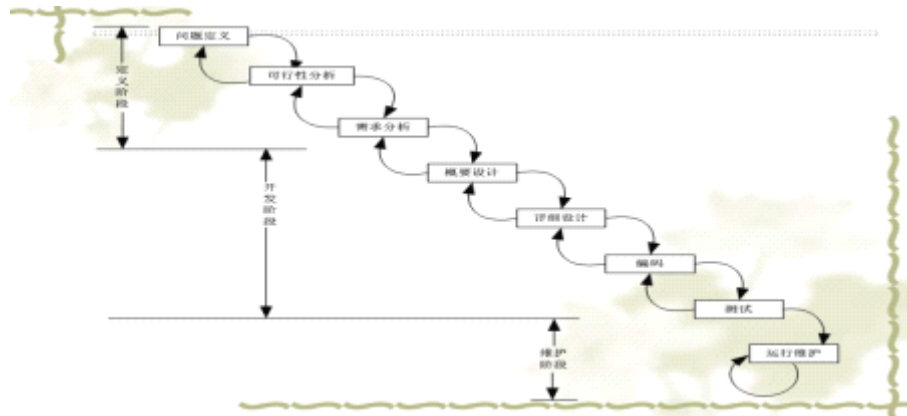
1. 技术措施

使用更好的软件开发方法和开发工具

2. 组织管理措施

软件开发不是某种个体劳动的神秘技巧，而应该是一种组织良好、管理严密、各类人员协同配合、共同完成的工程项目。

2 瀑布模型分为哪几个阶段？阶段之间有什么特点？请图示

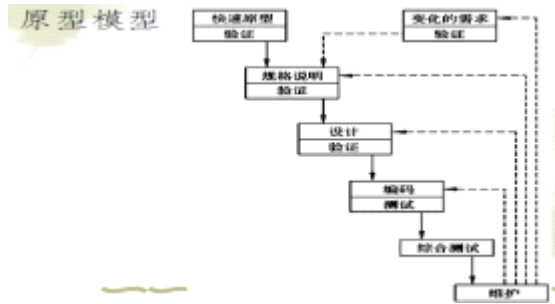


阶段间具有顺序性和依赖性这个特点有两重含义：①必须等前一阶段的工作完成之后，才能开始后一阶段的工作；②前一阶段的输出文档就是后一阶段的输入文档，因此，只有前一阶段的输出文档正确，后一阶段的工作才能获得正确的结果。

3 快速原型的基本思想是什么？请图示其过程模型

基本思想

在获取一组基本的需求定义后，利用高级软件工具的可开发环境，快速地建立一个目标系统的最初版本，并把它交给用户试用、补充和修改，再进行新的版本开发。反复进行这个过程，直到得出系统的“精确解”，即用户满意为止



4 软件项目可行性研究包括哪些内容？（第 2 讲）

1. 经济可行性 2. 技术可行性 3. 运行可行性 4. 法律可行性 5. 开发方案可行性

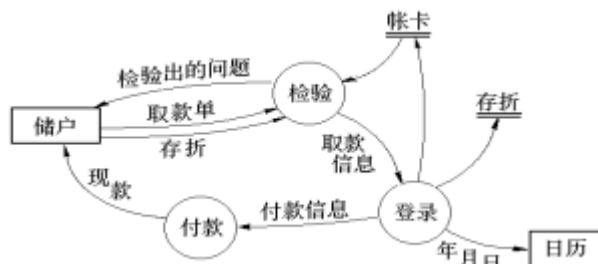
5 什么是需求分析

- ❖ **需求分析**是指开发人员要准确地理解用户的要求,进行细致的调查分析,将用户非形式化的需求陈述转化为完整的需求定义,再由需求定义转化为相应的**软件需求规格说明书**（即需求分析的结果）的过程。

需求分析过程：（获取用户需求，分析用户需求，编写需求文档，评审需求文档）

6 怎么画 ER 图 (第3讲)

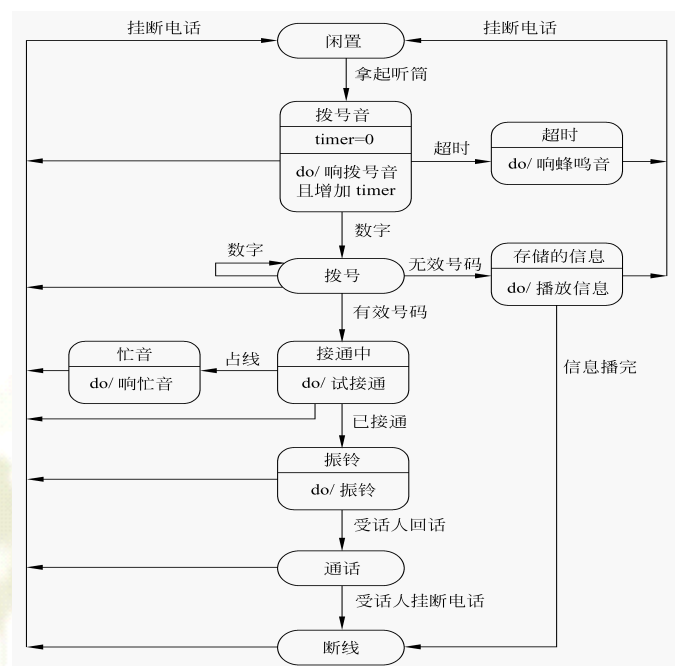
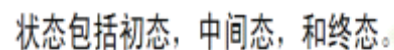
7 怎么画数据流图 (第 2 讲) (关注第 4 讲有案例)



(描述银行取款过程的数据流图)

8 怎么样画系统状态变迁图 (第3讲状态转换图)(第4讲状态图)

电话系统状态图



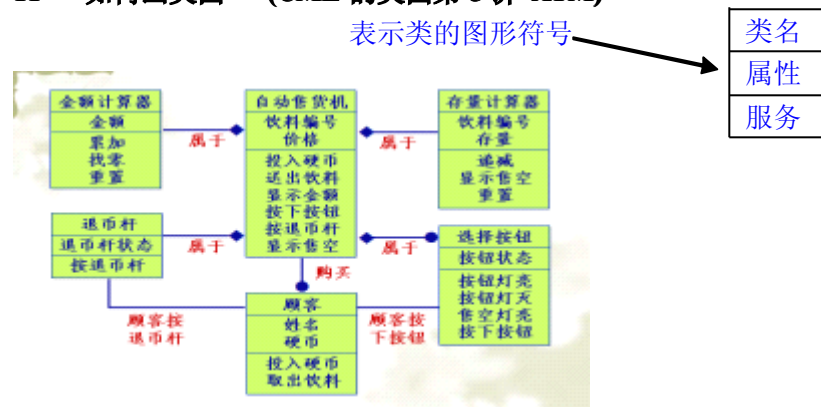
9 如何编写场景 (用例也叫场景) (第4讲)

- ❖ 寻找行为者
- ❖ 谁使用系统主要功能
- ❖ 谁借助系统完成日常工作
- ❖ 谁来维护系统
- ❖ 系统控制哪些设备
- ❖ 系统与其他系统交互
- ❖ 哪些人对本系统感兴趣

10 如何画用例图 (第4讲)

11 如何画类图 (UML 的类图第5讲 ATM)

表示类的图形符号



12 对象模型的5个层次是什么 (第5讲)

面向对象的3个模型: 对象模型、动态模型、功能模型;

5个层次(无序): (1)主题层 (2)类与对象层 (3)结构层 (4)属性层 (5)服务层

13 从需求陈述中提取对象模型的步骤 (第5讲)

- (1)筛选 (2)备选对象 (3) 确定类间关联 (4)筛选 (5)完善
(6)确定属性 (7) 识别继承关系 (8)反复修改

(1.确定对象类和关联 2.给类和对象添加属性 3.设计继承关系 4. 设计类中的操作)

14 面向对象设计的5个准则是什么 (第7讲)

- (1)模块化 (2)信息隐藏 (3)弱耦合 (4)强内聚 (5)可重用

15 软件开发中, 有哪些成分是可以重用的 (第7讲)

(代码重用、设计结果重用、分析结果重用)

项目计划、成本估计、体系结构、需求模型和规格说明、设计、源代码、
用户文档和技术文档、用户界面 (可占到60%的代码)、数据、测试用例

16 软件设计中, 一般的软件系统可以分解成哪四个子系统? (第7讲)

人机交互子系统、问题域子系统、任务管理子系统、数据管理子系统

17 界面设计中应该了解用户的哪些信息 (第8讲)

用户类型、使用系统欲达到的目的、特征(年龄、性别、受教育程度、限制因素等)、关键的成功因素(需求、爱好、习惯等)、技能水平、完成本职工作的脚本。

18 怎么样实现类间关联 (第9讲) (第5讲)

{ 1、由需求陈述中的动词词组表示关联关系。2、确定隐含的关联 3、根据问题域知识得到的关联 }

- ❖ 关联有两种可能的方式: 单向遍历和双向遍历。

- ❖ 用指针可以方便地实现单向关联。
- ❖ 实现双向关联有下列 3 种方法：
- ❖ (1) 只用属性实现一个方向的关联，当需要反向遍历时就执行一次正向查找。如果两个方向遍历的频率相差很大，而且需要尽量减少存储开销和修改时的开销。
- ❖ (2) 两个方向的关联都用属性实现。这种方法能实现快速访问，但是，如果修改了一个属性，则相关的属性也必须随之修改，才能保持该关联链的一致性。当访问次数远远多于修改次数时，这种实现方法很有效。
- ❖ (3) 用独立的关联对象实现双向关联。关联对象不属于相互关联的任何一个类，它是独立的关联类的实例。

19 如何用委托实现类的重用设计（第 9 讲）

如果你只想把继承作为实现操作共享的一种手段，则利用委托(**即把一个类对象作为另一类对象的属性，从而在两类对象间建立组合关系**) 也可以达到同样目的，而且这种方法更安全。使用委托机制时，只有有意义的操作才委托另一类对象实现，因此，不会发生不慎继承了无意义（甚至有害）操作的问题

20 为了提高编码的健壮性，应该遵循什么样的准则（第 10 讲）

- (1) 预防用户的操作错误 (2) 检查参数的合法性
- (3) 不要预先确定限制条件 (4) 先测试后优化

21 软件测试的目标及 6 条准则是什么（第 10 讲）

软件测试的目标或定义：

测试是为了**发现程序中的错误**而执行程序的过程；

（成功的测试是指发现了迄今为止没有发现的错误）

软件测试的 6 条准则：

- (1) 所有测试都应该能追溯到用户需求。 （以用户需求为中心）
- (2) 应该远在测试开始之前就制定出测试计划。（先制定测试计划）
- (3) 把 Pareto 原理应用到软件测试中。 （运用 Pareto 原理）
- (4) 应该从“小规模”测试开始，并逐步进行“大规模”测试。 （由小到大）
- (5) 穷举测试是不可能的。 （穷举法不可取）
- (6) 为了达到最佳的测试效果，应该由独立的第三方从事测试工作。（由独立的第三方来测试）

22 软件测试分为哪几步（第 10 讲）

- (1) 模块测试(2) 子系统测试(3) 系统测试(4) 验收测试(5) 平行运行

23 代码审查怎么样进行（第 11 讲）

审查小组最好由下述 4 人组成：

- (1) 组长，应该是一个很有能力的程序员，而且没有直接参与这项工程；
- (2) 程序的设计者；(3) 程序的编写者；(4) 程序的测试者。(5) 其他人员

审查之前，小组成员应该先研究设计说明书，力求理解这个设计。为了帮助理解，可以先由设计者扼要地介绍他的设计。

在审查会上由程序的编写者解释他是怎样用程序代码实现这个设计的，通常是逐个语句地讲述程序的逻辑，小组其他成员仔细倾听他的讲解，并力图发现其中的错误。

审查会上进行的另外一项工作，是对照类似于上一小节中介绍的程序设计常见错误清单，分析审查这个程序。当发现错误时由组长记录下，审查会继续进行(审查小组的任务是发现错误而不是改正错误) 。

24 解释单元测试的驱动模块和桩模块（第 11 讲）

模块并不是一个独立的程序，因此必须为每个单元测试开发“驱动模块”和 (或) “桩模

块”(存根模块)。

通常驱动程序也就是一个“主程序”，它接收测试数据，把这些数据传送给被测试的模块，并且印出有关的结果。

存根程序代替被测试的模块所调用的模块。因此存根程序也可以称为“虚拟子程序”。它使用被它代替的模块的接口，可能做最少量的数据操作，印出对入口的检验或操作结果，并且把控制归还给调用它的模。

25 什么是回归测试，alpha 测试 beta 测试（第 11 讲）

回归测试就是用于保证由于调试或其他原因引起的变化，不会导致非预期的软件行为或额外错误的测试活动。Alpha 测试由用户在开发者的场所进行，并且在开发者对用户的“指导”下进行测试。开发者负责记录发现的错误和使用中遇到的问题。总之，Alpha 测试是在受控的环境中进行的。Beta 测试由软件的最终用户们在一个或多个客户场所进行。

26 什么是白盒测试和黑盒测试（第 11、12 讲）

黑盒测试(功能测试): 已知产品应该具有的功能，可通过测试来检验是否每个功能都能正常使用；

白盒测试(结构测试): 已知产品的内部工作过程，可通过测试来检验产品内部动作是否按照规格说明书的规定正常进行。

(测试步骤: 1)模块测试 2) 子系统测试 3) 系统测试 4) 验收测试 5) 平行运行)

27 设计测试用例时，怎么样进行等价类划分和边界值分析（第 12 讲）

等价类划分法

概念

等价类划分法是把程序的输入域划分成若干部分（子集），然后从每个部分中选取少数代表性数据作为测试用例。每一类的代表性数据在测试中的作用等价于这一类中的其他值。

等价类划分法的应用

- 等价类是指某个输入域的子集合。在该子集中，各个输入数据对于揭露程序中的错误都是等效的，并合理地假定：测试某等价类的代表值就等于对这一类其它值的测试，因此，可以把全部输入数据合理划分为若干等价类，在每一个等价类中取一个数据作为测试的输入条件，就可以用少量代表性的测试数据，取得较好的测试结果。等价类划分可有两种不同的情况：有效等价类和无效等价类。
- 有效等价类：是指对于程序的规格说明来说是合理的、有意义的输入数据构成的集合。利用有效等价类可检验程序是否实现了规格说明中所规定的功能和性能。
- 无效等价类：与有效等价类的定义恰巧相反。
- 设计测试用例时，要同时考虑这两种等价类。因为，软件不仅要能接收合理的数据，也要能经受意外的考验。这样的测试才能确保软件具有更高的可靠性。

划分等价类的六大原则

- 在输入条件规定了取值范围或值的个数的情况下，则可以确立一个有效等价类和两个无效等价类
- 例：输入值是学生成绩。范围是0~100
- 在输入条件规定了输入值的集合或者规定了“必须如何”的条件的情况下，可确立一个有效等价类和一个无效等价类
- 在输入条件是一个布尔量的情况下，可确定一个有效等价类和一个无效等价类。布尔量是一个二值枚举类型，一个布尔量具有两种状态：true 和 false。
- 在输入数据的一组值（假定n个），并且程序要对每一个输入值分别处理的情况下，可确立n个有效等价类和一个无效等价类。
- 例：输入条件说明输入字符为：中文、英文、阿拉伯文三种之一，则分别取这三种这三个值作为三个有效等价类，另外把三种字符之外的任何字符作为无效等价类。
- 在输入数据必须遵守的规则的情况下，可确立一个有效等价类（符合规则）和若干个无效等价类（从不同角度违反规则）
- 在确知已划分的等价类中各元素在程序处理中的方式不同的情况下，则应再将该等价类进一步的划分为更小的等价类

将等价类转化成测试用例

- 按照[输入条件][有效等价类][无效等价类]建立等价类表，列出所有划分出的等价类
- 为每一个等价类规定一个唯一的编号
- 设计一个新的测试用例，使其尽可能地覆盖尚未被覆盖地有效等价类，重复这一步，直到所有的有效等价类都被覆盖为止
- 设计一个新的测试用例，使其仅覆盖一个尚未被覆盖的无效等价类，重复这一步，直到所有的无效等价类都被覆盖为止

边界值分析法

边界值分析法

概念

边界值分析法就是对输入或输出的边界值进行测试的一种黑盒测试方法。通常边界值分析法是作为对等价类划分法的补充。这种情况下，其测试用例来自等价类的边界。

边界值分析法的应用

- 根据大量的测试统计数据，很多错误是发生在输入或输出范围的边界上，而不是发生在输入/输出范围的中间区域。因此针对各种边界情况设计测试用例，可以查出更多的错误。
- 使用边界值分析方法设计测试用例，首先应确定边界情况。通常输入和输出等价类的边界，就是应着重测试的边界情况。应当选取正好等于，刚刚大于或刚刚小于边界的值作为测试数据，而不是选取等价类中的典型值或任意值作为测试数据。
- 边界值分析法与等价类分析法的区别：
- 边界值分析不是从某等价类中随便挑一个作为代表，而是使这个等价类的每个边界都要作为测试条件。
- 边界值分析不仅考虑输入条件，还要考虑输出空间产生的测试情况。
- 例：测试计算平方根的函数
- --输入：实数
- --输出：实数
- --需求说明：当输入一个0或比0大的数的时候，返回其正平方根；当输入一个小于0的数时，显示错误信息“平方根非负-输入值小于0”并返回0；库函数Print-Line可以用来输出错误信息。

• A等价类划分：

I.可以考虑作出如下划分：

- a、输入 (i)<0 和 (ii)>=0
- b、输出 (a)>=0 和 (b) Error

II.测试用例有两个：

- a、输入4，输出2。对应于 (ii) 和 (a)。
- b、输入-10，输出0和错误提示。对应于 (i) 和 (b)。

• B边界值分析：

划分(ii)的边界为0和最大正实数；划分(i)的边界为最小负实数和0。由此得到以下测试用例：

- a、输入 {最小负实数}
- b、输入 {绝对值很小的负数}
- c、输入 0
- d、输入 {绝对值很小的正数}
- e、输入 {最大正实数}

- 通常情况下，软件测试所包含的边界检验有几种类型：数字、字符、位置、重量、大小、速度、方位、尺寸、空间等。
- 相应地，以上类型的边界值应该在：最大/最小、首位/末位、上/下、最快/最慢、最高/最低、 最短/最长、 空/满等情况下，利用边界值作为测试数据

项	边界值	测试用例的设计思路
字符	起始-1个字符/结束+1个字符	假设一个文本输入区域允许输入1个到255个字符，输入1个和255个字符作为有效等价类；输入0个和256个字符作为无效等价类，这几个数值都属于边界条件值。
数值	最小值-1/最大值+1	假设某软件的数据输入域要求输入5位的数据值，可以使用10000作为最小值、99999作为最大值；然后使用刚好小于5位和大于5位的数值来作为边界条件。
空间	小于空余空间一点/大于满空间一点	例如在用U盘存储数据时，使用比剩余磁盘空间大一点（几KB）的文件作为边界条件。

内部边界值分析

- 在多数情况下，边界值条件是基于应用程序的功能设计而需要考虑的因素，可以从软件的规格说明或常识中得到，也是最终用户可以很容易发现问题的。然而，在测试用例设计过程中，某些边界值条件是不需要呈现给用户的，或者说用户是很难注意到的，但同时确实属于检验范畴内的边界条件，称为内部边界值条件或子边界值条件。

内部边界值条件主要有下面几种：

- 数值的边界值检验：计算机是基于二进制进行工作的，因此，软件的任何数值运算都有一定的范围限制。

项	范围或值
位 (bit)	0 或 1
字节 (byte)	0 ~ 255
字 (word)	0~65535 (单字) 或 0~4294967295 (双字)
千 (K)	1024
兆 (M)	1048576
吉 (G)	1073741824

- 其它边界值检验：在不同的行业应用领域，依据硬件和软件的标准不同而具有各自特定的边界值。如下列出部分手机相关的边界值：

硬件设备	范围或值
手机锂电池电压	工作电压：3.6~4.2V； 保护电压：2.5~3V不等
手机正常使用温度	-25° C~+60° C

- 基于边界值分析方法选择测试用例的原则

- 如果输入条件规定了值的范围，则应取刚达到这个范围的边界的值，以及刚刚超越这个范围边界的值作为测试输入数据。
- 例如，如果程序的规格说明中规定：“重量在10公斤至50公斤范围内的邮件，其邮费计算公式为……”。作为测试用例，我们应取10及50，还应取10.01,49.99,9.99及50.01等。
如果输入条件规定了值的个数，则用最大个数，最小个数，比最小个数少一，比最大个数多一的数作为测试数据。
例如，一个输入文件应包括1~255个记录，则测试用例可取1和255，还应取0及256等。
- 将规则1)和2)应用于输出条件，即设计测试用例使输出值达到边界值及其左右的值。
例如，某程序的规格说明要求计算出“每月保险金扣除额为0至1165.25元”，其测试用例可取0.00及1165.24，还可取-0.01及1165.26等。
再如一程序属于情报检索系统，要求每次“最少显示1条、最多显示4条情报摘要”，这时我们应考虑的测试用例包括1和4，还应包括0和5等。
- 如果程序的规格说明给出的输入域或输出域是有序集合，则应选取集合的第一个元素和最后一个元素作为测试用例。
- 如果程序中使用了一个内部数据结构，则应当选择这个内部数据结构的边界上的值作为测试用例。
- 分析规格说明，找出其它可能的边界条件。

28 代码行估算的方法，并用 Bohem 简单模型估算工作量，（第 13 讲）

为了使得对程序规模的估计值尽可能接近实际值，可以由多位有经验的软件工程师分别独立地作出估计。每个人都估计程序的最小规模 (a)、最大规模 (b) 和最可能的规模 (m)，分别算出这三类规模的平均值，和之后，再用下式计算程序规模的估计值：

$$L = \frac{\bar{a} + 4\bar{m} + \bar{b}}{6} \quad (7.1)$$

用代码行技术度量软件规模时，当程序较小时常用的单位是代码行数 (LOC)，当程序较大时常用的单位是千行代码数 (KLOC)。

Boehm 简单模型： $E = 3.2 * (KLOC)^{1.05}$ (指数幂)

其中，E 是以人月为单位的软件开发工作量

29 通过 putnam 模型进行开发时间的估算 (第 14 讲)

Putnam 模型 $T=2.4E^{1/3}$

其中, E 是开发工作量 (以人月为单位), T 是开发时间 (以月为单位)。

30 为什么说向一个延期的项目增加人手, 只能进一步延期 (第 14 讲)

经验告诉我们, 随着开发小组规模扩大, 个人生产率将下降, 以致开发时间与从事开发工作的人数并不成反比关系。

1、当小组变得更大时, 每个人需要用更多时间与组内其他成员讨论问题、协调工作, 因此增加了通信开销。

2、如果在开发过程中增加小组人员, 则最初一段时间内项目组总生产率不仅不会提高反而会下降。这是因为新成员在开始时不仅不是生产力, 而且在他们学习期间还需要花费小组其他成员的时间

31 Gantt 图和网络图 (第 14 讲)

32 软件开发 SQA 小组职责有哪些

SQA 小组的职责, 是辅助软件工程师以获得高质量的软件产品。其从事的软件质量保证活动主要是: 计划, 监督, 记录, 分析和报告。简而言之, SQA 小组的作用是, 通过确保软件过程的质量来保证软件产品的质量。

33 文档审查怎么做 (第 14 讲)

审查过程包括下述 5 个基本步骤:

(1)综述。由负责编写文档的一名成员向审查组综述该文档。在综述会结束时把文档分发给每位与会者。

(2)准备。评审员仔细阅读文档。最好列出在审查中发现的错误类型, 并按发生频率把错误类型分级, 以辅助审查工作。这些列表有助于评审员们把注意力集中到最常发生错误的区域。

(3)审查。评审组仔细走查整个文档。和走查一样, 这一步的目的也是发现文档中的错误, 而不是改正它们。通常每次审查会不超过 90 分钟。审查组组长应该在一天之内写出一份关于审查的报告。

(4)返工。文档的作者负责解决在审查报告中列出的所有错误及问题。

(5)跟踪。组长必须确保所提出的每个问题都得到了圆满的解决 (要么修正了文档, 要么澄清了被误认为是错误的条目)。必须仔细检查对文档所做的每个修正, 以确保没有引入新的错误。如果在审查过程中返工量超过 5%, 则应该由审查组再对文档全面地审查一遍。

34 软件配置管理主要有哪 5 项任务 (第 15 讲)

1. 标识软件配置中的对象 2. 版本控制 3. 变化控制 4. 配置审计 5. 状态报告

即 (标识对象、版本控制、变化控制、配置审计和状态报告)

35 CMM(能力成熟度模型)改进过程分哪 5 级 (第 15 讲)

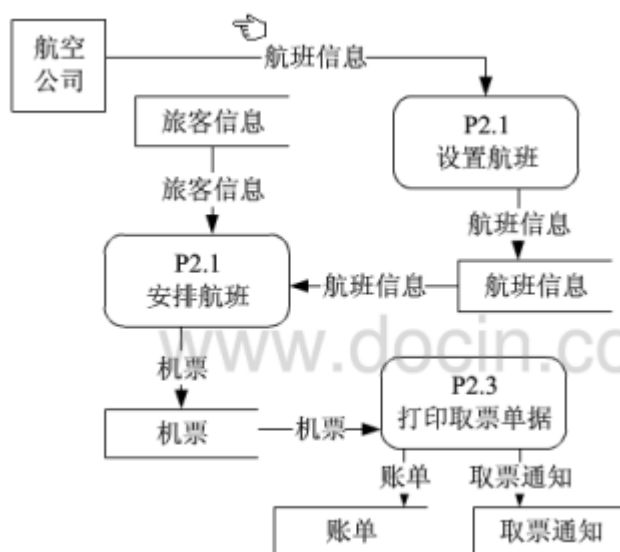
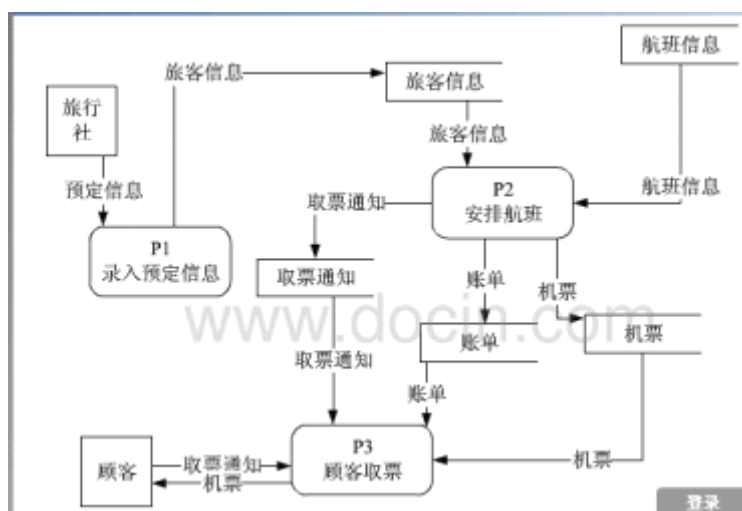
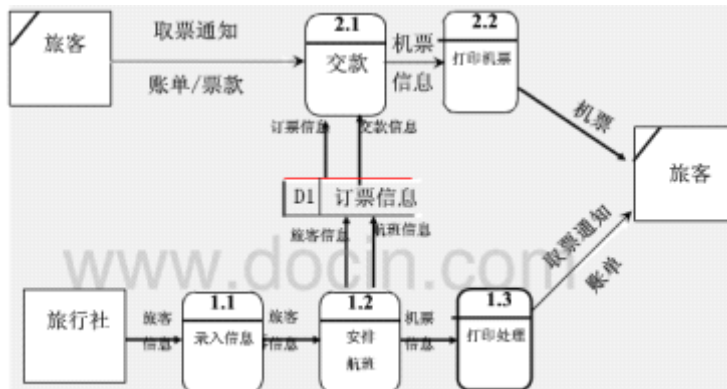
答案: 1、初始级 2、可重复级 3、已定义级 4、已管理级 5、优化级

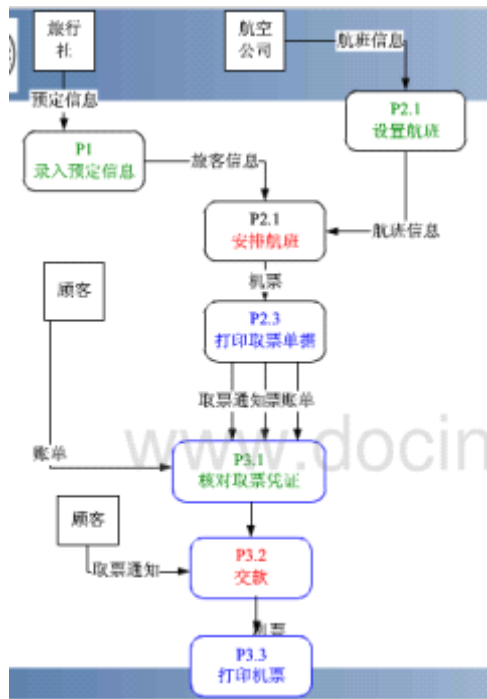
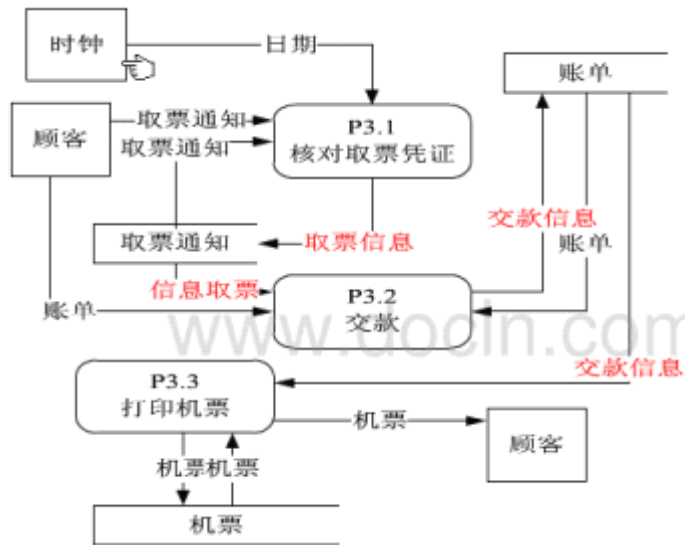
1 绘制教师, 学生, 选课的 ER 图

2 使用状态变迁图表示打印机空闲, 工作, 卡纸, 缺纸的状态变迁。

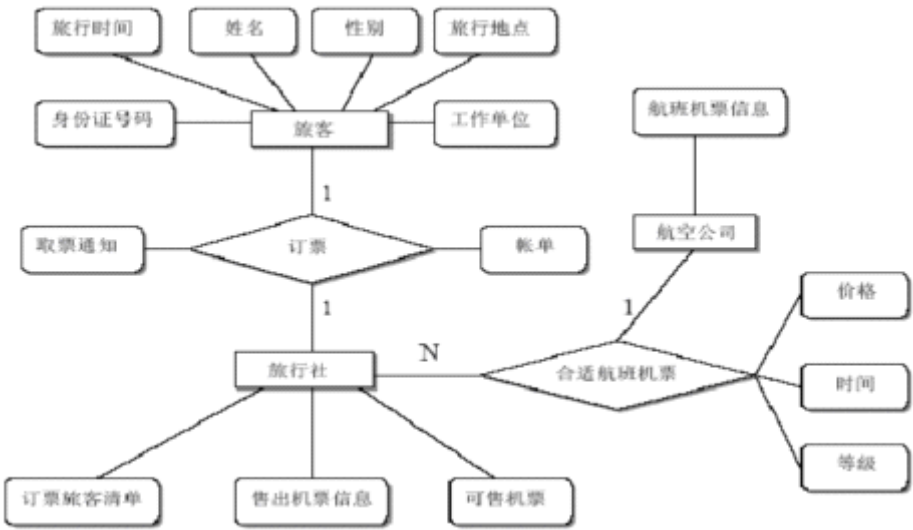
3 为方便旅客, 某航空公司拟开发机票预订系统, 旅行社把预订机票的旅客信息 (姓名, 性别, 工作单位, 身份证号码, 旅行时间, 旅行目的地等) 输入该系统, 系统安排航班, 印出取票通知和账单, 旅客在飞机起飞前一天凭取票通知和账单交款取票, 系统校对无误即印出

机票给旅客。问题：(1)绘制系统的数据流图 (2)建立系统的对象模型(应该是画类图)(3)建立系统的用例模型(画用例图)





ER 图



4 某程序规格说明有如下描述：某城市电话号码由三部分组成。它们的名称和内容分别是：
地区码：空白或三位数字；
前 缀：非 ‘0’ 或 ‘1’ 的三位数字；
后 缀：4 位数字。
假定被测程序能接受一切符合上述规定的电话号码，拒绝所有不符合规定的电话号码。
根据该程序的规格说明，**作等价类的划分**，并设计测试方案。

划分等价类

输入条件	有效等价类	无效等价类
地区码	1. 空白； 2. 三位数；	1. 有非数字字符； 2. 少于三位数字； 3. 多于三位数字。
前 缀	3. 从 200 到 999 之间的三位	4. 有非数字字符； 5. 起始位为‘0’； 6. 起始位为‘1’； 7. 少于三位数字； 8. 多于三位数字。
后 缀	4. 四位数字。	9. 有非数字字符； 10. 少于四位数字； 11. 多于四位数字。

设计测试方案

方 案	内 容			输 入	预期输出
	地区码	前缀	后缀		
1	空白	200~999 之间	四位数字	()276-2345	有效
2	三位数字	的三位数字	四位数字	(635)805-9321	有效
3	有非数字字符			(20A)723-4567	无效
4	少于三位数字			(33)234-5678	无效
5	多于三位数字			(5555)345-6789	无效
6		有非数字字符		(345)5A2-3456	无效
7		起始位为‘0’		(345)012-3456	无效
8		起始位为‘1’		(345)132-3456	无效
9		少于三位数字		(345)92-3456	无效
10		多于三位数字		(345)4562-3456	无效
11			有非数字字符	(345)342-3A56	无效
12			少于四位数字	(345)342- 356	无效
13			多于四位数字	(345)562-34567	无效

个人补充：

- ❖ 面向对象的方法开发系统模型包括：
 - ☞ 描述数据结构的对象模型
 - ☞ 描述控制结构的动态模型
 - ☞ 描述功能的功能模型

(结构化方法-ER 图，状态图，数据流图)

C. 成功的测试是发现了迄今尚未发现的错误的测试

45、单元测试是发现编码错误，集成测试是发现模块的接口错误，确认测试是为了发现功能错误，那么系统测试是为了发现（ ）的错误。

- ☞ A、接口错误 B、编码错误
C、性能、质量不合要求 D、功能错误

17.欲开发一个银行的活期存取款业务的处理系统：储户将填好的存/取款单和存折交给银行工作人员，然后由系统作以下处理：

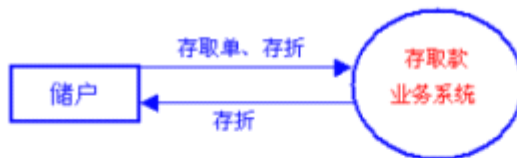
(1)业务分类处理：系统首先根据储户所填的存/取款单，确定本次业务的性质，并将存/取款单和存折交下一步处理；

(2)存款处理：系统将存款单上的存款金额分别记录在存折和帐目文件中，并将现金存入现金库；最后将存折还给储户；

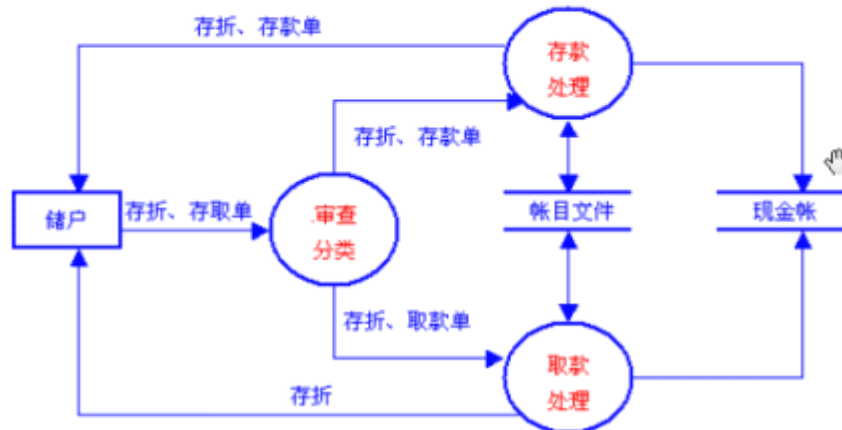
(3)取款处理：系统将取款单上的取款金额分别记录在存折和帐目文件中，并从现金库提取现金；最后将现金和存折还给储户。

绘制该系统的数据库图和数据流程图。

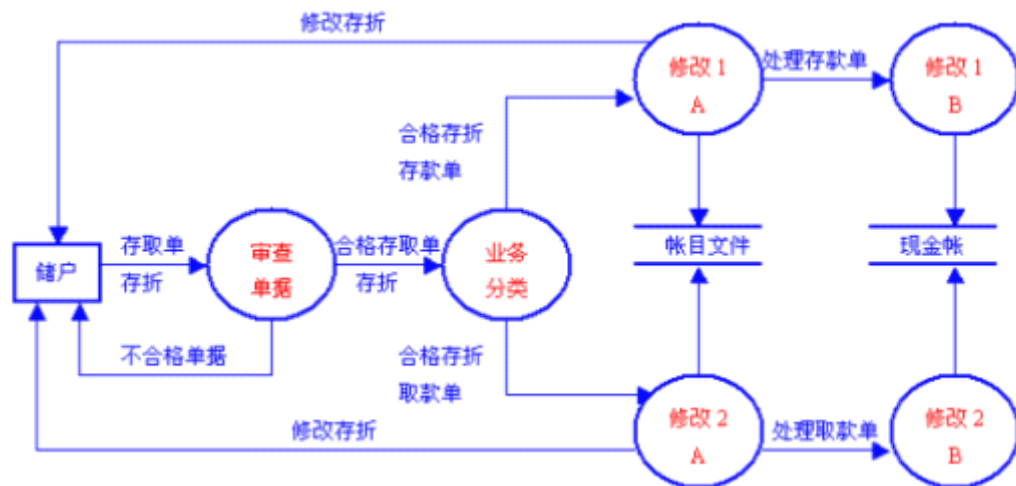
顶层



一层



二层



注：A—文件帐目及存折；B—现金帐

26. 等价分类法的测试技术采用的一般方法?举例说明?

- (1) 为每个等价类编号；
 - (2) 设计一个新的测试方案, 以尽可能多的覆盖尚未被覆盖的有效等价类, 重复这一步骤, 直到所有有效等价类被覆盖为止。
 - (3) 设计一个新的测试方案, 使它覆盖一个尚未被覆盖的无效等价类, 重复这一步骤, 直到所有无效等价类被覆盖为止。
- (例略)