目录

[1.第一章 2](#_Toc2144573597)

[1.概念，原因，典型表现 2](#_Toc799392325)

[2.消除软件危机的途径：对计算机软件深刻认识。 2](#_Toc472430050)

[3. 软件工程基本原理（7条） 2](#_Toc417625188)

[4. 软件工程三要素 2](#_Toc756575169)

[5.软件生命周期 2](#_Toc243364097)

[6可行性分析，需求分析，总体设计，详细设计，编码和单元测试（实现），综合测试 2](#_Toc510255636)

[7. 软件模型：瀑布，快速 3](#_Toc361827657)

[2.第二章 可行性研究 3](#_Toc1928477197)

[1.任务：是否值得解决（三方面）:经济，操作，技术 3](#_Toc1545556334)

[2.过程（自己看)（8条）（小题） 3](#_Toc1053339209)

[3.重要!数据流图 （组成，符号） 3](#_Toc235456088)

[5. 数据字典（概念就行） 3](#_Toc820837302)

[3. 第三章 需求分析 3](#_Toc148273214)

[1. 基本任务 3](#_Toc21456565)

[2.模型 3](#_Toc1431645179)

[3.结构化思想：面向数据流自顶向下 3](#_Toc1445120671)

[4. E-R图 3](#_Toc51720988)

[5. 状态转换图 3](#_Toc2140749004)

[4. 第五章 总体设计（第四章不看） 4](#_Toc72338821)

[1. 重要任务设计软件结构 4](#_Toc788201729)

[2. ！重要 设计原理 4](#_Toc941887803)

[3. 启发原则 4](#_Toc1396894725)

[4. 层次图，结构图（不出重点） 4](#_Toc1874678717)

[5. 面向数据流设计方法 4](#_Toc747430072)

[6.第六章 详细设计 4](#_Toc1442469211)

[1.基本控制结构 设计工具 4](#_Toc784438376)

[2.过程设计工具 4](#_Toc1061724341)

[3程序复杂度，流图（两种元素，节点，判定框，边） 4](#_Toc859256530)

[6. 第七章 实现 5](#_Toc492961429)

[1. 概念 5](#_Toc1799785438)

[2. 测试的目标 5](#_Toc856346479)

[3. 测试的方法：白，黑 5](#_Toc1292353754)

[4. 测试步骤，类型 5](#_Toc124731840)

[5. 编码 5](#_Toc1273971667)

[7. 第八章 维护 5](#_Toc2048928923)

[1. 概念，含义 5](#_Toc368095938)

[2. 维护的类型（改进，适应，） 5](#_Toc1784227303)

[3. 可维护性（确定可维护性的因素） 5](#_Toc263272932)

[8. 第九章 面向对象 5](#_Toc149089487)

[1. 要点 5](#_Toc1182299990)

[2. 定义，基本原则 5](#_Toc1316612142)

[3. 优点 5](#_Toc384545575)

[4. 三种模型（ 5](#_Toc2003137292)

[5. 类图，对象图，用例图，顺序图，UML 5](#_Toc1464885356)

[6. 三种模型关系 5](#_Toc406002140)

## 1.第一章 软件工程学概述

### 1.软件危机概念，原因，典型表现

概念：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。

典型表现：1.对软件开发成本和进度的估计常常很不准确。

2.用户对“已完成的”软件系统不满意的现象经常发生。

3.软件产品的质量往往靠不住。

4.软件常常是不可维护的。

5、软件通常没有适当的文档资料

6、软件成本在计算机系统总成本中所占的比例逐年上升。

7、软件开发生产率提高的速度，远远跟不上计算机应用迅速普及深入的趋势。

原因：1.与软件本身特点有关（不同于硬件）

2.软件开发与维护的方法不正确有关

需求分析，维护

### **2.消除软件危机的途径：**对计算机软件深刻认识。

1.依靠完整的程序代码，数据，文档

2.软件开发是多人团队分工，不是个人。

技术，方法

软件工具

如何消除

1. 对计算机软件有重要的认识
2. 使用开发软件的成功方法技术
3. 开发使用更好的软件工具

### 软件工程 基本原理（7条）

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、采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

### 软件工程三要素

方法，工具，过程

结构方法，面向对象方法：含义，定义

软件工程方法学

面向对象方法学（4要点），基本原则

### 5.软件生命周期

三阶段 定义，开发，维护

8个周期

1. 问题定义
2. 可行性研究
3. 需求分析
4. 总体设计
5. 详细设计
6. 编码和单元测试
7. 综合测试
8. 软件维护

### 6可行性分析，需求分析，总体设计，详细设计，编码和单元测试（实现），综合测试

### 软件模型：瀑布，快速

优点缺点

瀑布模型

阶段具有顺序性和依赖性

推迟实现的观点

质量保证的观点

优点：严格地规定了每个阶段必须提交的文档

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

快速原型模型:

名词解释:快速原型是快速建立起来的可以在计算机上运行的程序，它所能完成的功能 往往是最终产品能完成的功能的一个子集。

特点:不带反馈环的、软件开发基杜上是线在须序进行的

优点：较短时间满足用户需求

给用户学习过程

## 2.第二章 可行性研究

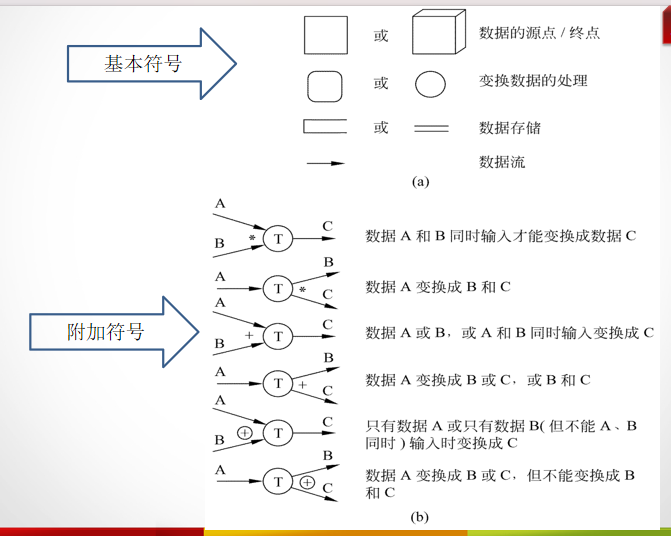
### 1.任务：用最小的代价在尽可能短的时间内确定问题是否能够解决

### （三方面）:经济，操作，技术

### 过程（自己看)（8条）（小题）

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. 进一步定义问题
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

### 3.重要!数据流图 （组成，符号）



各种符号等，应用（描述画图）,完成书上例题。 加工，实体，存储。（元素名）

### 数据字典（概念就行）

数据字典是关于数据的信息的集合，也就是对数据流图中包含的所有元素的定义的集合。

## 第三章 需求分析

### 基本任务

准确回答系统必须做什么 这个问题

### 2.模型

数据模型：E-R模型

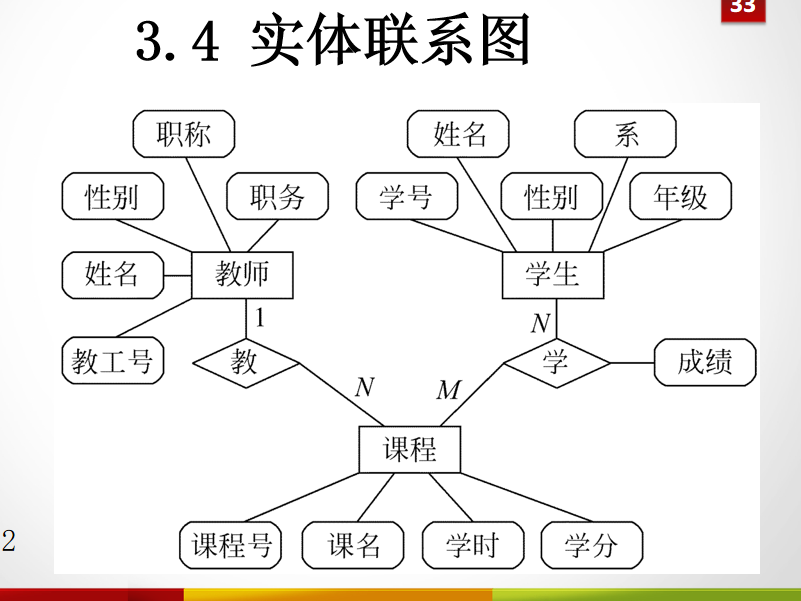
功能模型：DFD

行为模型：状态图

### 3.结构化思想：面向数据流自顶向下求精

### E-R图

实体含义，关系，属性



### 状态转换图

状态转换

## 第五章 总体设计（第四章不看）

任务：概括地说，系统应该如何实现

### 重要任务设计软件结构

### ！重要 设计原理

模块化，

模块是由边界元素限定的相邻程序元素（例如，数据说明，可执行的语句）的序列，而且有一个总体标识符代表它。模块是构成程序的基本构件。

模块化就是把程序划分成独立命名且可独立访问的模块，每个模块完成一个子功能，把这些模块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。

模块独立性

内聚性，耦合性

模块的独立程度可以由两个定性标准度量，这两个标准分别称为内聚和耦合。

尽量使用数据耦合，少用控制耦合和特征耦合，限制公共环境耦合的范围，完全不用内容耦合。

### 启发原则

扇出扇入：

扇出：一个模块直接控制(调用)的模块数目

扇入：一个模块被多少个上级模块直接调用的数目

作用域，控制域：不再控制怎么办（图5.2）

作用域定义

### 层次图，结构图（不出重点）

### 面向数据流设计方法

信息流：变化流，事物流（通过事务中心）

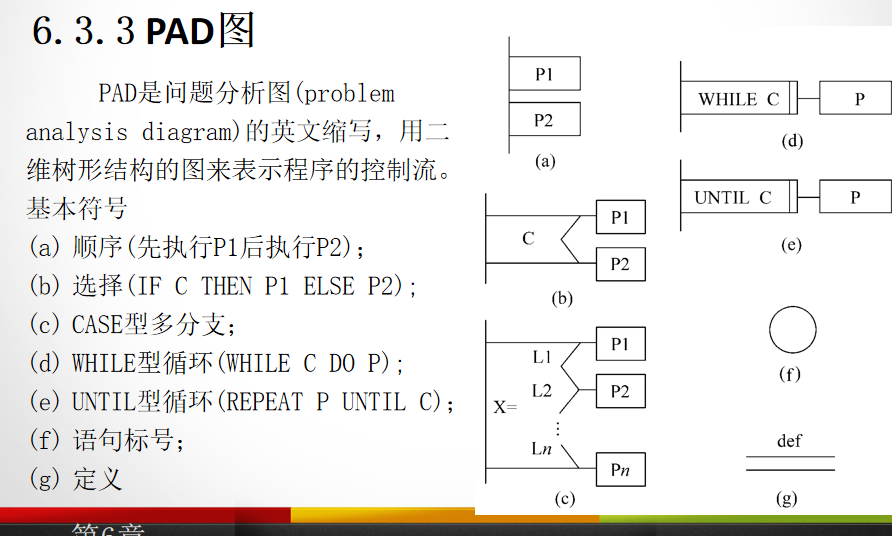
图5.20

## 6.第六章 详细设计

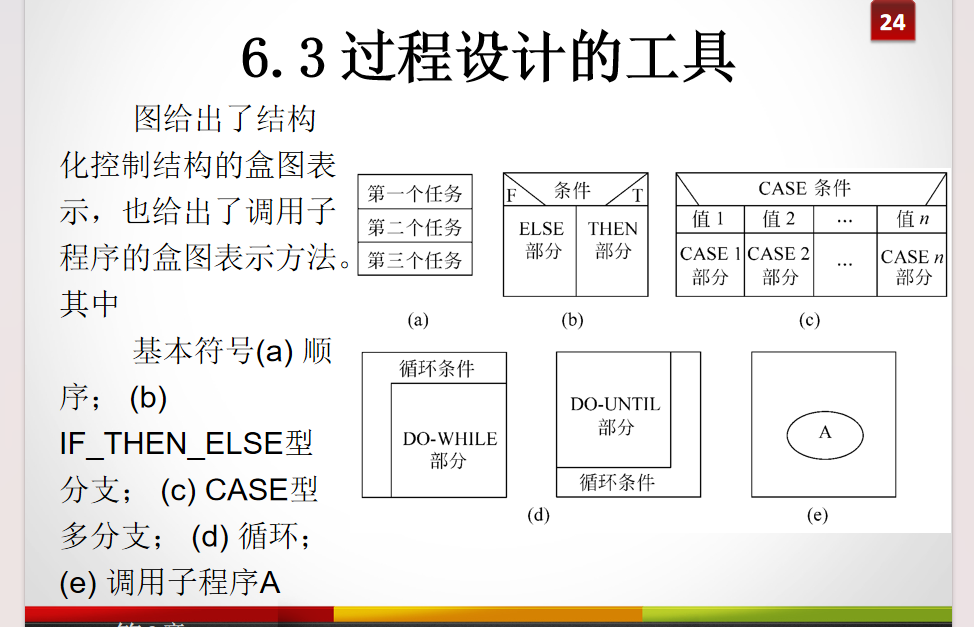
### 1.基本控制结构 设计工具

### 2.过程设计工具

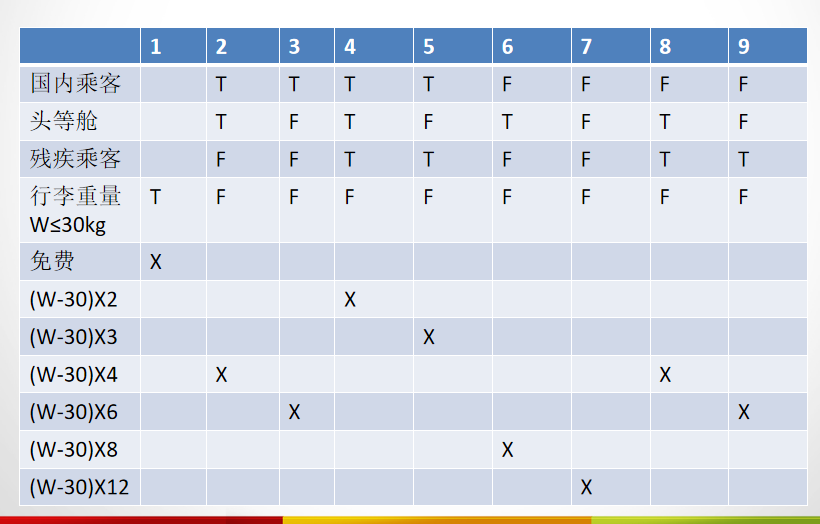
PAD图，



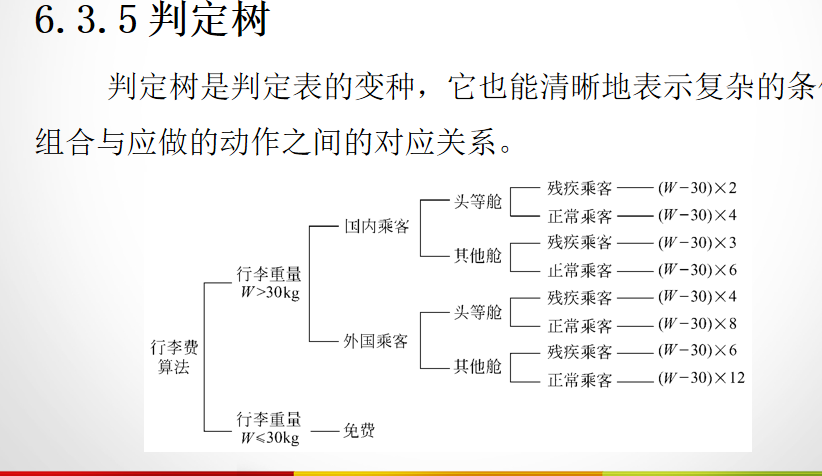
盒图，



判定表



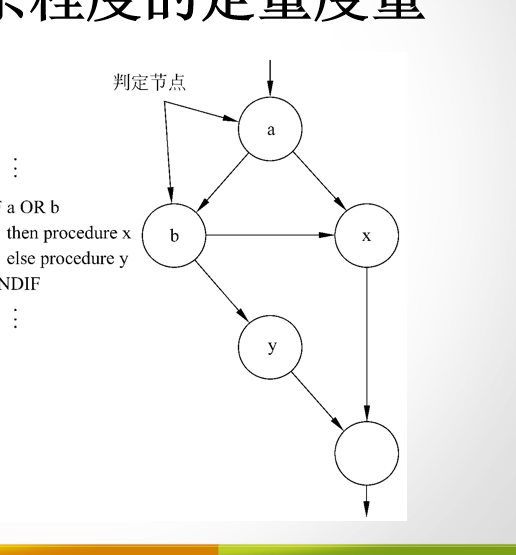
判定树 （理解例题）

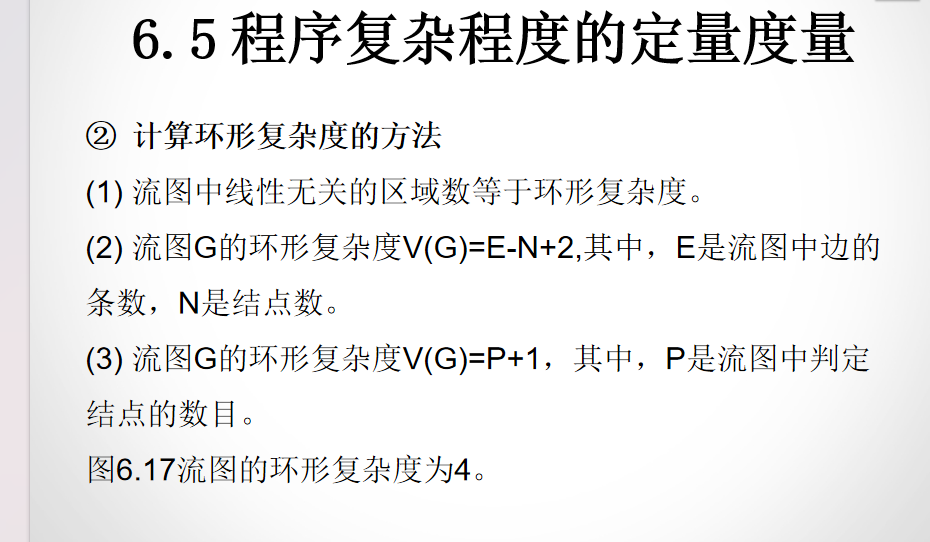


### 3程序复杂度，流图（两种元素，节点，判定框，边）

复杂度计算方法：3种







## 第七章 实现

### 概念

### 测试的目标

### 测试的方法：白，黑

### 测试步骤，类型

自顶下集成

自底向上集成

黑盒：

1等价划分

1. 边界值分析

3.错误推测

模块测试，单元测试，系统测试，子系统测试，集成测试，

### 编码

1. 集成测试

## 第八章 维护

### 概念，含义

### 维护的类型（改进，适应，）

1、改正性维护 17%~21%

适应性维护 18%~25%

3、完善性维护 50%-66%

4、预防性维护

### 可维护性（确定可维护性的因素）

可理解性

2、可测试性

3、可修修改性

4、可移植性

5 可重用性

## 第九章 面向对象

### 要点

### 定义，基本原则

### 优点

### 三种模型（

### 类图，对象图，用例图，顺序图，UML

### 三种模型关系