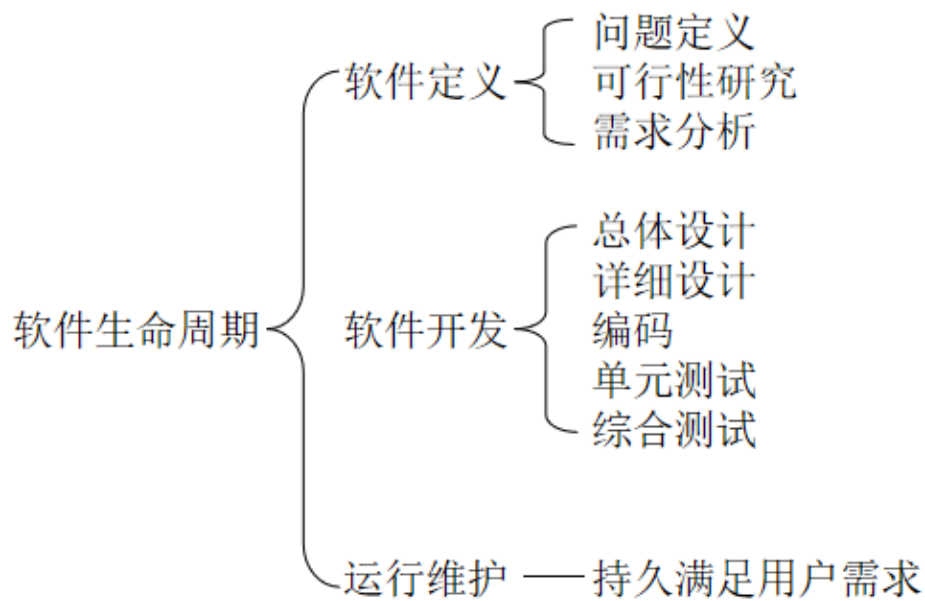


目录

1 软件开发工程基础.....	1
2 可行性分析.....	1
3 软件需求分析.....	2
4 软件设计.....	4
5 编程实现.....	5
6 软件测试.....	6

1 软件开发工程基础

软件生命周期



2 可行性分析

可行性分析的主要内容是对问题的定义，主要经过调研与初步概要设计，初步确定软件项目的规模和目标，明确项目的约束和限制，并导出软件系统的逻辑模型；然后从此模型出发，确定若干可供选择的主要软件系统初步研发方案。

可行性分析主要包括 5 个方面：技术可行性、经济可行性、社会可行性、开发方案可行性和运行可行性。其中**最主要的工作是前三项**。

技术可行性主要分析在特定条件下，技术资源、能力、方法等方面的可用性及其用于解决软件问题的可能性和现实性。**技术可行性分析的内容**包括：对新软件功能的具体指标、运行环境及条件、响应时间、存储速度及容量、安全性和可靠性等要求；确定现有资源条件下的技术风险及项目能否实现等。其中的资源包括已有的或可以取得的硬件、软件和其他资源，现有技术人员的技术水平和已有的工作基础。

经济可行性，主要从资源配置的角度衡量软件项目的实际价值，分析研发软件所需成本和项目开发成功后带来的经济收益。**研发软件的成本**有 4 个部分组成：（1）购置并安装软硬件及其有关网络等设备的费用，考虑服务器、主机及其外围设备等硬件费用，并注意软件及其隐私和潜在的费用。（2）软件系统开发费用。（3）软件的安装、管理、配置、运行和维护等。（4）推广及用户使用与人员培训等费用。

社会可行性，开发新软件之前还应该兼顾对法律、经济及发展变化等各种社会因素的约束要求。

运行可行性，新软件系统的运行可行性主要包含下面几个方面：（1）原业务与新软件功能及其流程的相近程度和差异。（2）业务处理的专业化程度、功能、性能、安全性、可靠性及接口等（当前系统与其他系统进行数据交换）。（3）对各种用户操作方式及其具体使用的要求。（4）新软件界面的接受程度和操作的便捷程度。（5）用户的具体实际应用能力及存在的问题。

开发方案可行性，包括资源和时间等可行性分析，具体表现在如下四个方面。（1）以正常的运作方式，开发软件项目并投入市场的可行性分析。（2）需要人力资源、财力等预算。（3）软件、硬件及研发设备等物品资源等设备预算情况。（4）组织保障及时间进度保障分析等。

3 软件需求分析

软件需求通常分为 4 个层次：

（1）**业务需求**：反映组织机构或客户对系统和产品高层次的目标要求，他们在项目视图与范围文档中予以说明。

（2）**用户需求**：是从用户的角度描述系统完成任务或者是用户期望的产品属性。用户需求文档，用以描述用户使用产品要完成的任务。

（3）**功能需求**：主要描述系统所提供的功能或服务，即定义系统的主要功能、系统的输入输出信息、系统的约束。

（4）**非系统功能需求**：作为系统需求的补充，主要描述那些与系统的具体功能无关，但与系统的总体特性相关的特征，如安全性、可靠性、响应时间、可移植性、可重用性等。

需求分析的任务：借助当前系统的逻辑模型导出目标系统的逻辑模型，解决目标系统“做什么”的问题。在需求分析阶段的最后，需要提交较为详细的数据流图、数据字典和需求分析报告。

需求分析的方法有一个叫做：**结构化分析**。

结构化分析导出的模型包括，数据模型、功能模型和行为模型，这些模型以“数据字典”为核心，数据字典描述了软件使用的所有数据对象。如下图所示。其中，实体-关系图（ER 图）作为数据建模的基础，描述数据对象实体及其关系；数据流图作为功能建模的基础，描述数据怎样转换及其转换功能；状态转换图作为行为建模的基础，表示系统的各种行为状态以及状态间的转换方式。



图 3-3 结构化分析模型

实体-联系图（E-R 图）：由实体、联系和属性构成。

联系：指实体间的相互联系。实体之间的联系可主要划分为 3 类。

一对一（1: 1），例如，一个学院有一名院长。

一对多（1: n），例如，每个出版社出版多本书，但是每本书只能出自一个出版社。

多对多（n: n），例如，学生与课程之间，一个学生可以选择多门课程，每门课程可以有多个学生。

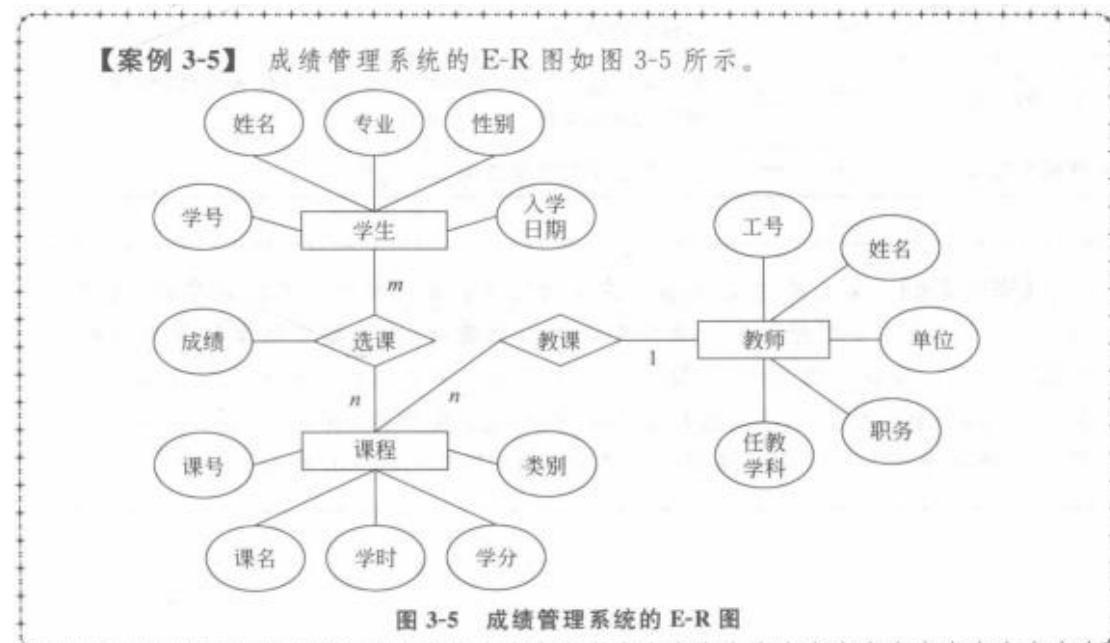
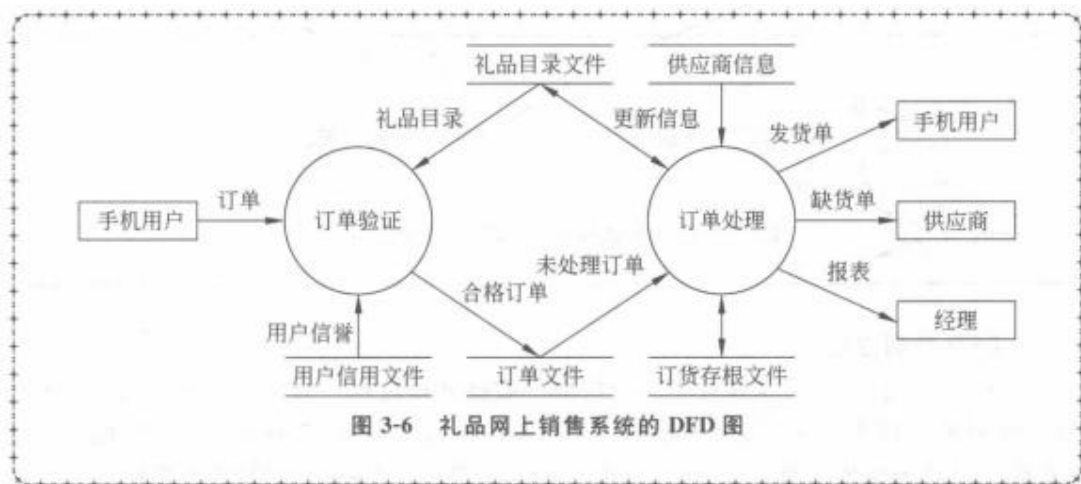


图 3-5 成绩管理系统的 E-R 图

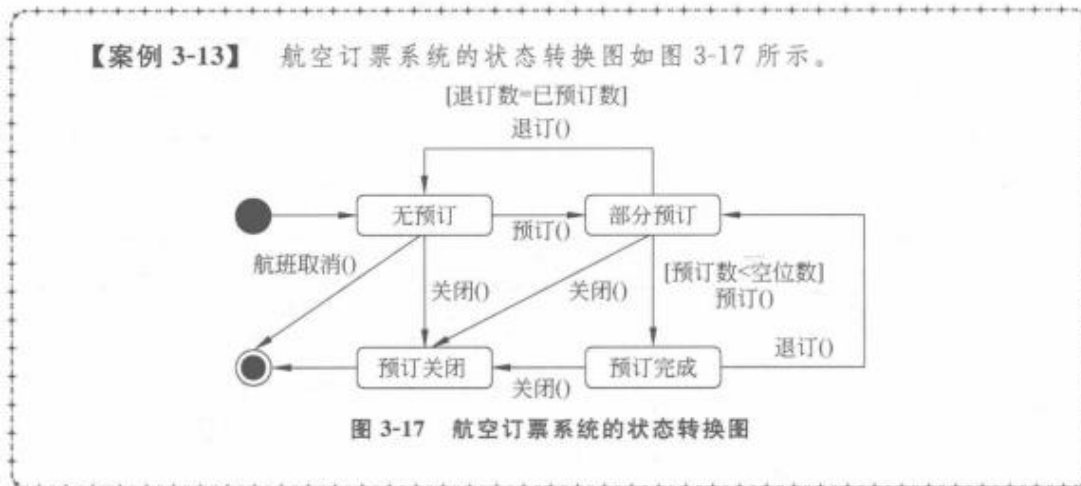
数据流图（DFD）：是一种图形化的系统模型，在一张图中将新软件系统建模作为输入、处理、输出和数据存储。DFD 图的描述符号由四种，起点（或终点）、数据流连线、数据加功处理、输入输出文件。

表 3-5 DFD 描述符号

名 称	图 例	说 明
起点(或终点)		数据流的起点或终点,表示数据源和数据库
数据加工/处理		表示对流到此处的数据进行加工或处理,即对数据的算法分析与科学计算
输入输出文件		表示输入输出文件,说明加工/处理前的输入文件,记录加工/处理后的输出文件,也可为单线
数据流连线		表示数据流的流动方向



状态转换图 (STD): 通过描绘系统的状态及其引起状态变化的事件表示系统的行为, 还指明了作为特定事件的结果系统将做出哪写动作。



4 软件设计

软件设计总体目标: 将软件需求分析阶段确定的各种功能对应了逻辑模型, 转换为具体的物

理模型。（描述软件“如何做”，即“实现方案”的物理过程）

软件设计分为两个阶段：总体设计和详细设计。

总体设计：在此阶段应确定软件系统的具体实现方案、给出软件的模块结构、编写总体设计文档等。

详细设计：是对总体设计的一个具体化过程，确定组成模块及联系、处理过程、数据库设计、网络设计、界面设计等。

总体设计：利用合适的设计方法和工具，将一个复杂的软件系统按功能划分模块，概要设计其功能、性能、可靠性和接口，以及模块间的调用关系。

模块的内聚与耦合。追求**高内聚低耦合**。

软件总体设计的方法，**结构化开发方法**（SD 方法）。SD 方法总体设计过程需要从 DFD 图导出初始的模块结构图，先要分析 DFD 图的类型，对于不同的类型，采用不同的技术将其转换为初始的模块结构图。常将 DFD 图分为两种类型：**中心变换型和事务处理型**。

详细设计：根据总体设计提供的文档，对各个模块给出详细的过程性描述及其他具体设计等，完成相关文档及实现方案。

详细设计的工具。系统流程图，是一种常见的算法表示工具。

程序流程图 (Program Flow Chart) 也称为**程序框图**，是一种常用的算法表达工具。其中常用的**程序流程图基本符号**如图 4-13 所示。



图 4-13 流程图基本符号

数据库设计：包括概念结构设计（设计 E-R 模型）、逻辑结构设计（数据库表结构）、物理结构设计。

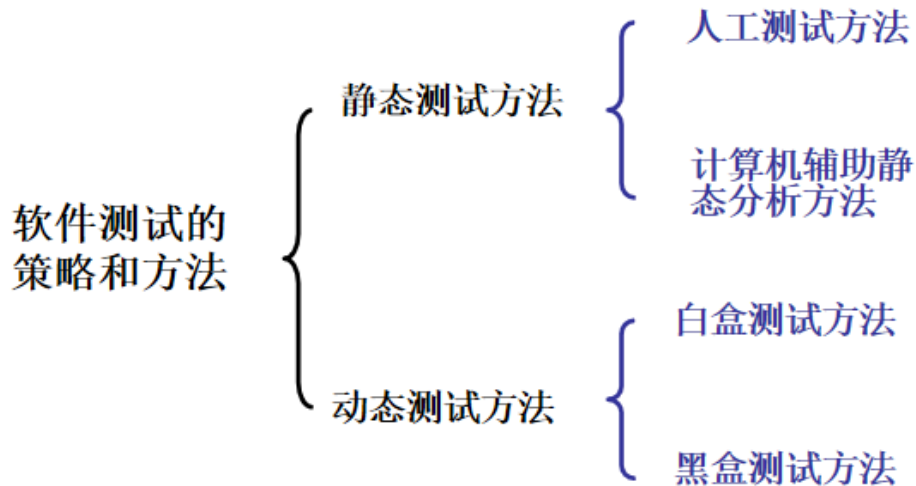
界面设计：用户界面应具备的特征：菜单、对话框、窗口、问题描述语言、数据表格、图形与图标等。用户界面设计的原则：（1）页面的合适性。（2）简便易操作。（3）便于交互控制。

（3）媒体组合适当。

5 编程实现

略

6 软件测试



软件测试：在规定的条件下对软件系统进行检测运行操作，以发现软件程序错误，衡量软件质量。

软件测试的分类。

从**功能上**分为：系统软件测试和应用软件测试。系统软件测试的主要任务是发现问题(bug)。应用软件测试的主要任务的发现功能、性能、可靠性、接口和界面等不符合项。

按**步骤**分：单元测试、集成测试、有效性（确认）测试和系统（整体）测试。

软件测试方法。

黑盒测试法：也称功能测试或黑箱测试。指测试人员只需被测如软件的界面和接口外部情况，不必考虑程序内部的逻辑结构和特性，只根据程序的需求分析，测试其功能是否符合。

常见的黑盒测试技术方法如下：

1 等价类划分法（等价分配）

把所有可能的输入数据（有效的和无效的）划分成若干个等价的子集（称为等价类别或等价区间），使得每个子集中的一个典型值在测试中的作用与这一子集中所有其它值的作用相同。

等价类别或等价区间是指测试相同目标或者暴露相同软件缺陷的一组测试用例

如何划分等价类？

- 有效等价类（合理等价类）
- 无效等价类（不合理等价类）

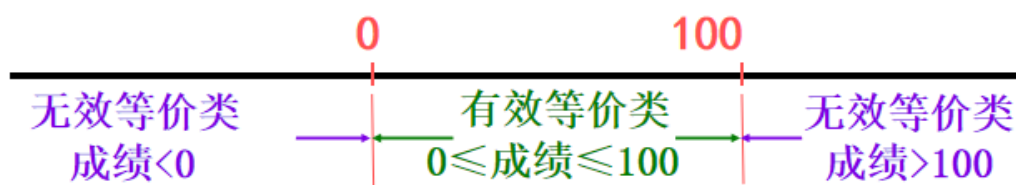
划分等价类的标准：

- 覆盖
- 不相交
- 代表性

划分等价类的规则

- (1) 如果输入条件规定了取值范围，可定义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



- (2) 如果输入条件代表集合的某个元素，则可定义一个有效等价类和一个无效等价类。

划分等价类的规则：

- (3) 如规定了输入数据的一组值，且程序对不同输入值做不同处理，则每个允许的输入值是一个有效等价类，并有一个无效等价类(所有不允许的输入值的集合)。

例：输入条件说明学历可为：专科、本科、硕士、博士四种之一，则分别取这四个值作为四个有效等价类，另外把四种学历之外的任何学历作为无效等价类

- (4) 如果规定了输入数据必须遵循的规则，可确定一个有效等价类（符合规则）和若干个无效等价类（从不同角度违反规则）。
- (5) 如已划分的等价类各元素在程序中的处理方式不同，则应将此等价类进一步划分成更小的等价类。

用等价类划分法设计测试用例步骤：

- (1) 形成等价类表，每一等价类规定一个唯一的编号；
- (2) 设计一测试用例，使其尽可能多地覆盖尚未覆盖的有效等价类，重复这一步骤，直到所有有效等价类均被测试用例所覆盖；
- (3) 设计一新测试用例，使其只覆盖一个无效等价类，重复这一步骤直到所有无效等价类均被覆盖；

例：某报表处理系统要求用户输入处理报表的日期，日期限制在2003年1月至2008年12月，即系统只能对该段期间的报表进行处理，如日期不在此范围内，则显示输入错误信息。

系统日期规定由年、月的6位数字字符组成，前四位代表年，后两位代表月。

如何用等价类划分法设计测试用例，来测试程序的日期检查功能？

第一步：等价类划分

“报表日期”输入条件的等价类表

输入条件	有效等价类	无效等价类
报表日期的类型及长度	6位数字字符 (1)	有非数字字符 (4) 少于6个数字字符 (5) 多于6个数字字符 (6)
年份范围	在2003~2008之间 (2)	小于2003 (7) 大于2008 (8)
月份范围	在1~12之间 (3)	小于1 (9) 大于12 (10)

第二步：为有效等价类设计测试用例

对表中编号为1, 2, 3的3个有效等价类用一个测试用例覆盖：

测试数据	期望结果	覆盖范围
200306	输入有效	等价类 (1) (2) (3)

(1) 6位数字字符
(2) 年在2003~2008之间
(3) 月在1~12之间

第三步：为每一个无效等价类设至少设计一个测试用例

测试数据	期望结果	覆盖范围
003MAY	输入无效	等价类(4)
20035	输入无效	等价类(5)
2003005	输入无效	等价类(6)
200105	输入无效	等价类(7)
200905	输入无效	等价类(8)
200300	输入无效	等价类(9)
200313	输入无效	等价类(10)
 不能出现相同的测试用例		<div>本例的10个等价类至少需要8个测试用例</div>

例：对招干考试系统“输入学生成绩”子模块
设计测试录入准考证号的测试用例

准考证号数据格式定义：共6为数字组成，其中
第一位为专业代号：1-行政专业，2-法律专业，3-财经专业
后5位为考生顺序号，编码范围为：

行政专业准考证号码为：110001~111215

法律专业准考证号码为：210001~212006

财经专业准考证号码为：310001~314015

准考证号码的等价类划分

有效等价类：(1) 110001 ~ 111215
(2) 210001 ~ 212006
(3) 310001 ~ 314015

无效等价类：(4) - ∞ ~ 110000
(5) 111216 ~ 210000
(6) 212007 ~ 31000
(7) 314016 ~ + ∞

等价类划分即把输入空间分解成一系列子域，软件在一个子域内的行为应是等价的。

软件错误分为两类：

计算错误
域错误

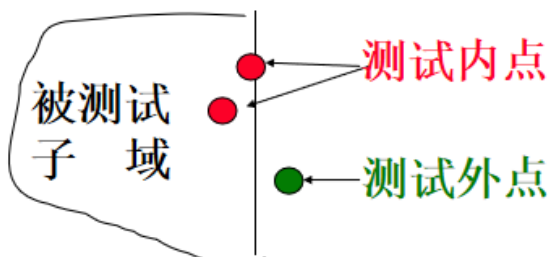
◆针对计算错误的测试方法

◆针对域错误的测试方法：测试域边界划定的正确性

2 边界值分析法

边界值分析法与等价类划分法区别

- (1) 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。
- (2) 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



软件边界与悬崖很类似

如果在悬崖峭壁边可以自信地安全行走，平地就不在话下。

如果软件在能力达到极限时能够运行，那么在正常情况下就不会出什么问题。

测试边界线

- 测试临近边界的合法数据, 以及刚超过边界的非法数据.
- 越界测试通常简单地加1或很小的数(对于最大值)和减1或很小的数(对于最小值).

“报表日期（6位数字字符）”边界值分析法测试用例

输入条件	测试用例说明	测试数据	期望结果	选取理由
报表日期的类型及长度	1个数字字符	5	显示出错	仅有1个合法字符
	5个数字字符	20035	显示出错	比有效长度少1
	7个数字字符	2003005	显示出错	比有效长度多1
	有1个非数字字符	2003. 5	显示出错	只有1个非法字符
	全部是非数字字符	MAY—	显示出错	6个非法字符
	6个数字字符	200305	输入有效	类型及长度均有效
日期范围	在有效范围边界上选取数据	200301	输入有效	最小日期
		200812	输入有效	最大日期
		200300	显示出错	刚好小于最小日期
		200813	显示出错	刚好大于最大日期
月份范围	月份为1月	200301	输入有效	最小月份
	月份为12月	200312	输入有效	最大月份
	月份<1	200300	显示出错	刚好小于最小月份
	月份>12	200313	显示出错	刚好大于最大月份