

目录

实验一 问题定义和可行性研究 (2)	1
实验二 需求分析(2)	5
实验三 总体设计(2)	7
实验四 详细设计(4)	8
实验五 编码实现及系统测试(6)	9
附录一：实验说明	12
一、实验安排方式	
二、实验环境	
三、实验要求	
四、实验成绩的评分方法和标准	
附录二：实验题目	13
附录三：实验报告范例	13

实验一 问题定义和可行性研究

(2)

一. 目的与要求

【目的】

- 1、确定课题熟悉软件开发环境，
- 2、掌握常用工具软件的使用方法。

【要求】

- 1、学习软件开发的组织和管理，将项目开发各阶段的任务明确。
- 2、熟悉相应的软件开发环境。
- 3、熟悉 Visio 的工作环境及组成；掌握 Visio 软件绘制图表的基本操作。

二. 实验基本原理

1、Visio 简介

Visio 是一个图表绘制软件，它有助于创建、说明和组织复杂设想、过程与系统的业务和技术图表。使用 Visio 创建的图表能够将信息形象化。生产与运营管理中涉及到的项目管理、质量管理、业务流程等内容，通过应用 VISIO 软件绘制相关图表，能够以清楚简明的方式有效地交流信息，提高了相关工作的效率和质量。

(1) 启动和退出 VISIO

- 1 执行【开始】→【程序】→【Microsoft Office】→【Microsoft Office Visio】菜单命令，即可启动 Visio。
- 1 直接单击视窗的开关按钮或选择主菜单中的 File/Exit 即可退出 Visio。

(2) 项目日程

- 1 PERT 图：计划评审技术 (PERT) 图用于规划、分析和监控项目。应用 Visio 软件绘制 PERT 图在项目早期组织任务、建立时间框架和阐释任务相关性方面很有用。
- 1 甘特图：甘特图是显示活动及持续时间的条形图。应用 Visio 软件绘制甘特图可用于安排、计划和管理项目。

(3) 业务进程

- 1 TQM 图：使用 Visio 软件的“TQM 图”模板可为全面质量管理项目创建流程图，有助于对当前流程和理想化的流程进行比较，并理解流程中各步骤是如何协同工作的。
- 1 基本流程图：使用 Visio 软件创建基本流程图可以展示过程、分析进程、指示工作流或信息流、跟踪成本和效率等。
- 1 因果图：因果图显示导致或影响特定情况的所有因素。使用 Visio 软件的“因果图”模板，有助于分析一个特定结果的主要原因和次要原因。

(4) 组织结构图

- 1 组织结构图：使用 Visio 软件的“组织结构图”模板，能够以图表形式表示组织的等级结构中人员之间、操作之间、职能之间以及活动之间的相互关系。

(5) Visio 画图步骤

- 1 选择一个所要的样板；
- 1 拖入一个原件：将模板中的原件拖入绘图页中，形成图形；
- 1 连接一些图形：将相互之间有关系的图形连接起来；
- 1 加上一些文字说明：给所有图形包括连接图形加上必要的文字说明；
- 1 对绘图文件进行美化处理和保存。

2、可行性研究

(1) 可行性研究

这是在软件项目计划阶段应该做的事情，包括四个方面的研究：

§ 经济可行性：进行成本 / 效益分析。从经济角度判断系统开发是否“合算”。

§ 技术可行性：进行技术风险评价。从开发者的技术实力、以往工作基础、问题的复杂性等出发，判断系统开发在时间、费用等限制条件下成功的可能性。

§ 法律可行性：确定系统开发可能导致的任何侵权、妨碍和责任。

§ 方案的选择：评价系统或产品开发的几个可能的候选方案。最后给出结论意见。

(2) 经济可行性

分析员需要进行成本 / 效益分析。所谓成本，包括：① 购置并安装软、硬件及有关设备的费用；② 系统开发费用；③ 系统安装、运行及维护的费用；④ 人员培训费用。而效益是指：① 系统为用户增加的收入或为用户节省的开支，这是有形的效益；② 给潜在用户心理上造成的影响，这是无形的效益。它可以转化为有形的效益。

(3) 技术可行性

分析员需要根据系统的功能、性能需求，建立系统模型。然后对此模型进行一系列的试验、评审和修改。最后由项目管理人员作出是否进行系统开发的决定。

如果开发技术风险很大，或者模型演示表明当前采用的技术和方法不能实现系统预期的功能和性能，或者系统的实现不支持各子系统的集成，则项目管理人员可以作出停止系统开发的决定。

(4) 方案的选择

分析员考虑问题解决的方案。一般采用将一个大而复杂的系统分解为若干个子系统的办法来降低解的复杂性。如何进行系统分解、如何定义各子系统的功能、性能和界面，实现方案不唯一。可以采用折衷的方法，反复比较各个方案的成本 / 效益，选择可行的方案。

3、可行性研究报告

可行性报告的形式可以有多种，但最重要的内容应当有：

- I. 项目背景：① 问题描述 ② 实现环境 ③ 限制条件
- II. 候选方案：① 候选系统的配置 ② 最终方案的选择标准
- III. 系统描述：① 系统的简要说明
- IV. 经济可行性（成本—效益分析）：① 经费概算 ② 软件成本估算 ③ 预期的经济效益
- V. 技术可行性（技术风险评价）：① 技术实力 ② 已有工作基础 ③ 设备条件

VI. 法律可行性：① 系统开发可能导致的侵权，违法和责任

VII. 用户使用可行性：① 用户单位的行政管理，工作制度 ② 使用人员的素质

VIII. 其它与项目有关的问题：① 其它方案介绍 ② 未来可能的变化

IX. 项目实施计划及进度安排

项目实施计划内容包括规定每项任务完成的起止时间、任务完成的标志、各项任务中参与的人数、工作量和各个任务之间的衔接情况，完成各项任务所需的工具资源和数据资源。

进度计划的内容包括：

- I 任务划分。
- I 工作量分配。
- I 工程进度安排。
- I 定义结果与责任。

可行性研究报告首先由项目负责人审查（审查内容是否可靠），再上报给上级主管审阅（估价项目的地位）。从可行性研究应当得出“行或不行”的决断。当然，在以后的开发阶段，还要其它“行还是不行”的决定。

三. 实验步骤

- 1、根据选择的课题，管理整个项目的实施过程，并合理调整资源和负责项目全局；针对项目的实施，熟悉相应的软件开发工具的使用环境。
- 2、准备参考资料和阅读相关的国家有关软件开发的标准文档。
- 3、使用 Microsoft Visio 来设计基本流程图模型。(以下四选三)
 - I 上机操作实验 1：框图
 - I 上机操作实验 2：基本流程图
 - I 上机操作实验 3：PERT 图
 - I 上机操作实验 4：甘特图
- 4 根据选择的课题，针对项目任务，编制可行性研究报告和项目实施计划。

四. 实验结果验收

- 1、确定项目题目
- 2、完成实验报告——问题定义及可行性分析报告

五、思考题

- 1、可行性研究和软件项目计划主要完成什么工作？
- 2、你认为项目开发计划中的最重要的问题是什么。
- 3、你认为项目计划怎么对软件开发有意义？

六、参考文献

- [1] 郑人杰，软件工程(21 世纪高等学校计算机规划教材:名家系列)，人民邮电出版社，2009.11
- [2] 系统自带的“Project 入门教程”。
- [3] 软件设计文档国家标准
- [4] MS-Project2000 参考教程

实验二 需求分析 (2)

一、目的与任务

【目的】

确定项目的可实施性，在此基础上完成系统的逻辑功能模型的建立。

【要求】

- 1、采用不同的软件开发技术，完成对项目的分析过程，给出系统的用例模型以及规格说明书。
- 2、熟悉用例图的基本功能和使用方法。

二、实验基本原理

1、软件需求概述

软件需求是指开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式功能规约(需求规格说明)的过程。软件需求虽处于软件开发过程的开始阶段，但它对于整个软件开发过程以及软件产品质量是至关重要的。

2、软件需求的任务

软件需求的主要任务就是要通过软件开发人员与用户的交流和讨论，准确地获取用户对系统的具体要求。在正确理解用户需求的前提下，软件开发人员还需要将这些需求准确地以文档的形式表达出来，作为设计阶段的依据。软件需求阶段结束时需要提交的主要文档是需求规格说明书。

用户对系统的需求通常可分为如下两类：

功能性需求：主要说明了待开发系统在功能上实际应做到什么，是用户最主要的需求。通常包括系统的输入、系统能完成的功能、系统的输出及其他反应。

非功能性需求：从各个角度对所考虑的可能的解决方案的约束和限制。主要包括：过程需求(如交付需求、实现方法需求等)、产品需求(如可靠性需求、可移植性需求、安全保密性需求等)和外部需求(如法规需求、费用需求等)等。

3、软件需求的步骤

软件需求阶段的工作，可分为几个步骤进行：

(1) 通过调查研究，获取用户的需求。

软件开发人员通过认真细致的调查研究，获得进行系统分析的原始资料。需求信息的获取可来源于阅读描述系统需求的用户文档；对相关软件、技术的市场调查；对管理部门、用户的访问咨询；对工作现场的实际考察等。

(2) 去除非本质因素，确定系统的真正需求。

对于获取的原始需求，软件开发人员需要根据掌握的专业知识，运用抽象的逻辑思维，找出需求间的内在联系和矛盾，去除需求中不合理和非本质的部分，确定软件系统的真正需求。

(3) 描述需求，建立系统的逻辑模型。

建立软件需求模型是需求分析的核心工作，它通过建立需求的多种视图，揭示出需求的不正确、不一致、遗漏和冗余等更深的问题。

(4) 书写需求说明书，进行需求复审。

需求阶段应提交的主要文档包括需求规格说明书、初步的用户手册和修正后的开发计

划。为了保证软件开发的质量，对软件需求阶段的工作要按照严格的规范进行复审，从不同的技术角度对该阶段工作做出综合性的评价。复审即要有用户参加，也要有管理部门和软件开发人员参加。

三. 实验步骤

- 1、到相关单位进行需求分析。
- 2、综合利用 Internet 网和相关书籍整理并完善需求分析。
- 3、完成需求规格说明书。

四、实验结果验收

完成实验报告——需求分析报告

五、思考题

- 1、需求分析在软件开发中真的有那么重要吗？
- 2、怎样组织对该工作的评审？

六. 参考文献

- [1] 郑人杰，软件工程(21 世纪高等学校计算机规划教材:名家系列)，人民邮电出版社，2009.11
- [2] UML 与 Rational Rose 讲义。

实验三 总体设计

(2)

一、目的与任务

【目的】

在实验二的基础上完成系统总体分析与设计 并给出相应的规格说明书。

二、实验要求

根据所选课题，完成软件结构设计。采用Visio等工具绘制软件结构图、软件层次图，并将系统数据流图转化成合理的软件结构图。

三、实验步骤

- 1、在实验二的基础上，采用Visio工具绘制软件结构图。
- 2、采用Visio或者Word工具绘制软件层次图。
- 3、分析系统数据流图，并转化成合理的软件结构图，并用Visio画出结构图。
- 4、进行系统的数据库设计。
- 5、完成总体设计报告。

四、实验结果验收

完成实验报告——总体设计报告

五、思考题

- 1、系统设计和需求分析的关系是什么？两者必须先后关联吗？
- 2、怎样组织对设计阶段工作的评审？

六、参考文献

[1] 郑人杰，软件工程(21 世纪高等学校计算机规划教材:名家系列)，人民邮电出版社，2009.11

实验四 详细设计

(4)

一、目的与任务

【目的】

在实验三的基础上完成系统详细设计，并给出相应的规格说明书。

二、实验要求

熟练掌握几种常用的软件详细设计工具，如程序流程图、盒图、PAD图、判定表和判定树。并能把给定的软件问题描述转化为过程设计结果，同时进行环形复杂度计算，判断结构化设计结果的复杂性。

三、实验步骤

- 1、在实验三的基础上，采用详细设计工具描述详细设计结果。
- 2、采用Visio画出程序流程图。
- 3、采用类C代码写出对应的伪代码。
- 4、将程序流程图转化为流图，然后计算它的McCabe环形复杂度。
- 5、完成详细设计报告。

四、实验结果验收

完成实验报告——详细设计报告

五、参考文献

[1] 郑人杰，软件工程(21 世纪高等学校计算机规划教材:名家系列)，人民邮电出版社，2009.11

实验五 编码实现级系统测试

(6)

一、目的与任务

【目的】

将详细设计转化为基于某种程序设计语言的可运行的程序代码，并完成对系统的测试。

【要求】

- 1、理解程序设计语言选择的标准。
- 2、了解编程的基本原则。
- 3、熟练掌握一种程序设计语言，将设计转化为可运行的程序代码。
- 4、了解用户说明手册和使用手册的编写。
- 5、熟悉单元测试、综合测试、确认测试 系统测试的方法，掌握测试用例的设计方法
- 6、能够正确运用软件测试的过程与方法，提高软件测试的实践能力，并给出测试报告。

二. 实验基本原理

1、软件实现概述

软件实现是软件产品由概念到实体的一个关键过程，它将详细设计得到的处理过程的描述转换为基于某种计算机语言的并且最终可以运行的程序代码。虽然软件的质量取决于软件设计，但是规范的程序设计风格对后期的软件维护带来不可忽视的影响。

2、程序设计语言的选择

通常选择程序设计语言时优先考虑高级语言，而不是低级语言(主要是汇编语言)。这是因为高级语言明显优于低级语言。

高级语言的选择可以参照以下标准。

① 理想标准

- I 为了使程序容易测试和维护以减少软件的总成本，所选用的高级语言应该有理想的模块化机制，以及可读性好的控制结构和数据结构。
- I 为了便于调试和提高软件可靠性，应该使编译程序能够尽可能多地发现程序中的错误。
- I 为了降低软件开发和维护的成本，选用的高级语言应该有良好的独立编译机制。

②实用标准

- I 语言自身的特性
- I 软件的应用领域
- I 软件开发的环境
- I 软件开发的方法
- I 算法和数据结构的复杂性
- I 软件可移植性要求
- I 软件开发人员的知识

目前在软件实现中使用面向对象语言非常普遍。到底应该选用面向对象语言还是非面向对象语言，关键不在于语言功能强弱。选择面向对象语言的关键因素，是语言的一致性的表达能力、可重用性及可维护性。开发人员在选择面向对象语言时，除了考虑上述的实用标准

以外，还应该着重考虑以下一些实际因素：

- | 可重用性。
- | 类库和开发环境。
- | 将来能否占主导地位。
- | 其他因素。

3、编码风格

编码风格又称程序设计风格或编程风格。编码风格实际上指编程的基本原则。

一个公认的、良好的编程风格可以减少编码的错误，减少读程序的时间，从而提高软件的开发效率。为了做到这一点，应该遵循下述一些原则：

- | 源程序文档化
- | 数据说明：在编写程序时，要注意数据说明的风格。
- | 语句构造：构造的语句要简单、直接，不要为了提高效率而使语句更为复杂。
- | 输入和输出：输入 / 输出的方式和格式应当尽量作到对用户友好，尽可能方便用户的使用。
- | 效率：选择良好的设计方法才是提高程序效率的根本途径，设计良好的数据结构与算法，都是提高程序效率的重要方法。

由于面向对象的程序设计语言所具有的特殊性，面向对象编程原则，除了遵循上述编程的基本原则之外，还包括为适应面向对象方法所特有的概念(如继承性)而必须遵循的一些新原则：

- | 提高可重用性
- | 提高可扩充性
- | 提高健壮性

总之，在编码时要善于积累编程经验，培养和学习良好的编程风格，使程序清晰易懂，易于测试与维护 从而提高软件的质量。

4、软件测试概述

Grenford J. Myers 在《The Art Of Software Testing》一书中的观点常被许多人作为软件测试的目的或定义：

- | 软件测试是为了发现缺陷而执行程序的过程。
- | 测试是为了证明程序中有错误，而不是证明程序中无错误。
- | 一个好的测试用例指的是它可能发现至今尚未发现的缺陷。
- | 一次成功的测试指的是发现了新的软件缺陷的测试。

测试的目的是想以最少的时间和人力找出软件中潜在的各种错误和缺陷。测试只能尽可能多的查找出程序中错误，而不能证明程序中没有错误。

5、软件测试的基本原则

- | 不完全原则：不完全原则表明测试是不完全的，穷举测试是不可能的。
- | 免疫性原则：软件缺陷具有免疫性，测试人员完成的测试越多，其免疫能力就越强，寻找更多软件缺陷也就更加困难。
- | 全程测试原则：全程测试原则要求软件测试不仅存在于完成程序之后，而应该跨越整个软件开发流程。
- | 80 / 20 原则：80 / 20 原则是指 80% 的软件缺陷存在于软件 20% 的空间里，软件缺陷具有空间聚集性。

6、软件测试方法

软件测试方法之所以没能完全标准化和统一化，主要原因是因为软件产业产品到软件测试有各式各样的软件。但是目前仍有很多各样软件测试方法都基本可用的常用概念和方法。我们这里介绍的思路和方法就是可用于多数应用程序的测试。有六个有关软件测试方法

的基本概念是很重要的：白箱测试，黑箱测试，有效用例和无效用例，边界条件以及等价类测试。

白箱测试或白盒测试（White-box testing 或 glass-box testing）是通过程序的源代码进行测试而不使用用户界面。这种类型的测试需要从代码句法发现内部代码在算法，溢出，路径，条件等等中的缺点或者错误，进而加以修正。

黑箱测试或黑盒测试（Black-box testing）是通过使用整个软件或某种软件功能来严格地测试，而并没有通过检查程序的源代码或者很清楚地了解该软件或某种软件功能的源代码程序具体是怎样设计的。测试人员通过输入他们的数据然后看输出的结果从而了解软件怎样工作。通常测试人员在进行测试时不仅使用肯定出正确结果的输入数据，而且还会使用有挑战性的输入数据以及可能结果会出错的输入数据以便了解软件怎样处理各种类型的数据。

边界条件（Boundary Cases）：环绕边界值的测试。通常意味着最大值，最小值或者所设计软件能够处理的最长的字符串等等。比如说某软件字体的字号支持范围是：从 8 到 72。那么边界测试用例应该包括：小于 8，等于 8，等于 72 和大于 72。

等价类（equivalent classes）：等价类测试用例指的是如果有很多测试用例执行再多也不会找到新的中的缺陷。因为虽然输入和输出结果有所不同，但是它们都通过同样的软件的源代码路径。通常只要一个源代码程序的路径是用于处理一定数值范围内的所有数值，那么除了边界值以外，在边界值范围以内的所有数值一般都属于等价类。因为如果软件程序能正确处理一个值，也就意味着该程序能正确处理在这个范围内的除了边界值以外的其他任何有效输入值。我们来用以上软件字体的字号来举例说明。软件支持的字号范围是：从 8 到 72。那么 8 和 72 之间的所有支持的字号都可以被认为是等价类的测试用例。再比如：测试超链接时两个用例 <http://www.yahoo.com/> 和 <http://www.yahoo2.com/> 也是等价类的测试用例

三．实验步骤

- 1、选用一种或几种程序设计语言，编程实现系统。
- 2、针对实验项目，设计测试用例
- 3、开展测试，并做好测试记录。
- 4、编写软件测试报告 and 用户使用手册。

四、实验结果验收

- 1、上交可以运行的系统及源代码。
- 2、完成《系统实现及软件测试报告》

五、思考题

- 1、怎样合理选择软件开发的工具？
- 2、怎样进行用户说明手册和使用手册的编写。
- 1、怎样看待测试的重要性。
- 2、怎样选择测试的技术和方案。
- 3、比较逻辑测试和基本路径测试的不同和联系。
- 4、怎样组织对测试阶段工作的评审？

六．参考文献

[1] 郑人杰，软件工程(21 世纪高等学校计算机规划教材:名家系列)，人民邮电出版社，2009.11

附录一：实验说明

一、实验安排方式

- 1、每人一题，选定开发项目。
2. 必须按照《软件工程实验指导书》附录中给定的文档规范标准提供项目文档。
3. 题目自定或采用附录二中的题目。
4. 软件开发的方法 为结构化方法。

二、实验环境:

根据所选择的软件项目来配置实验环境，一般可以选择以下三种之一：

1. 单机模式

操作系统：Windows 等

开发工具：C 语言或 Java

2. 客户机/服务器模式（C/S）

操作系统：Windows 等

数据库管理系统：Access, SQL Server, Oracle8i 5

开发工具：VB6.0, Dephi6.0, PB7.0, C++Builder, VC++, C#, VB.Net

3. Web 服务器模式（B/S）

操作系统：Windows

数据库管理系统：Access, SQL Server, Oracle8i

开发工具：C#, ASP.Net, Java

三、实验要求

1、预习

要求对实验的名称、目的、原理、仪器设备、数据记录方式、实验项目的调查研究、分析设计、测试修改等进行必要的准备。

2、实验操作

要求实验操作准确，实验现象明显，数据记录及时、数据处理和测试修改准确，如出现违章操作，拼凑，抄袭数据等情况，则应严格扣分。

3. 实验纪律

学生应遵守实验室纪律，爱护仪器设备，实验物品，保持实验室环境卫生。如出现下列情况则应严格扣分：

- a) 实验迟到或未完成实验擅自离开实验室。
- b) 操作违章或不守纪律，损坏公物。
- c) 不讲卫生，影响实验室环境。
- d) 实验完成后，未做好整理工作

四、实验成绩的评分方法和标准

1、实验成绩实行优秀、良好、中等、及格、不及格五级记分，或者采用百分制记分。

2、实验成绩的评定采用实验报告、作品（软件系统和建模）和考勤三项来考核，然后取比例值。具体比例实验报告 40%，作品 40%，考勤 20%，合计 100%。

3、文档部分实验评价标准

不能按要求或不能按时完成，成绩评为“不及格”（60 分以下）；

能按软件工程规范要求完成文档，软件调试、测试、运行通过，成绩评为“及格”（60—74 分）；

全部文档质量较好，软件有一定实用价值，成绩评为“良”（75—89 分）；

文档质量好，软件有难度，能反映同学较强能力，成绩评为“优”（90—100 分）

为了保证实验的质量，要求同学们对实验内容进行预习，重视对实验内容的理解和体会，认真完成每次“实验小结”，“实验小结”作为实验成绩的主要评价成分。

附录二：实验题目

学生管理系统	图书借阅系统	人事管理系统
网络商店	机房管理系统	工资管理系统
银行储蓄管理系统		商店管理系统
机票订购系统	试题管理系统	图书分销系统
科研成果管理系统	教务管理系统	大学排课系统
在线订餐系统	在线学习系统	实体商店管理系统

附录三：实验报告范例

实验三 需求分析报告

（题目与指导书一致）

实验目的与任务

目的：确定项目的可实施性，在此基础上完成系统的逻辑功能模型的建立。

任务：采用不同的软件开发技术，完成对软件开发项目的需求分析过程，给出系统的逻辑功能模型，数据字典以及规格说明书。

实验内容及要求

明白项目的业务流程图，绘制数据流图，功能分析图，编写数据字典，数据加工处理的描述，软件系统流程设想图（新系统模型），需求规格说明书。

实验结果（可续页）

思考题

- 1、需求分析在软件开发中真的有那么重要吗？
- 2、分析系统流程图，流程图和数据流图的区别和各自的特点。
- 3、怎样写符合规范的数据流图和数据词典？
- 4、怎样组织对该工作的评审？

实验小结：