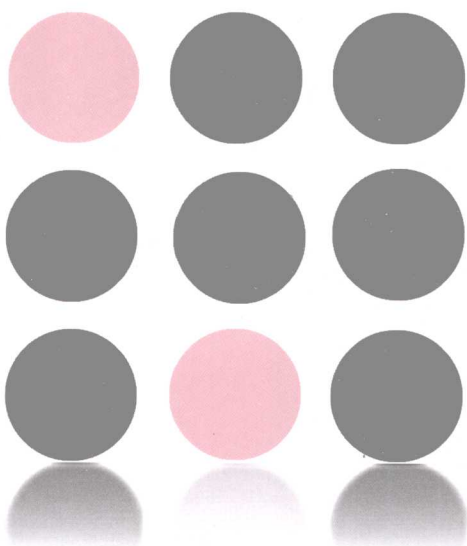


 高等院校计算机专业课程综合实验系列规划教材



丛书主编 何钦铭 陈根才

胡兰青 翁 恺 陈 奇 著

左保河 主 审

面向对象程序设计 课程设计



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

TP312/2883D

2008

浙江大学魏绍相计算机教材建设基金资助
高等院校计算机专业课程综合实验系列规划教材

面向对象程序设计课程设计

胡兰青 翁 恺 陈 奇 著

左保河 主审



ZHEJIANG UNIVERSITY PRESS

浙江大学出版社

内 容 提 要

本书是配合面向对象程序设计课程的课程设计指导教材,内容涵盖C++语言的重要基本知识、STL编程初涉、Windows编程入门等。通过对每个案例的相关背景知识的回顾、题目解析与实现要点的分析,循序渐进地启发学生完成课程设计,培养学生实际分析问题、编程和动手的能力,提高学生的综合素质。

本书所附光盘内容为案例的源程序。

本书适合作为高等院校面向对象程序设计课程设计指导用书,也可供广大计算机编程人员参考,具有很高的实用参考价值。

图书在版编目(CIP)数据

面向对象程序设计课程设计 / 胡兰青, 翁恺, 陈奇著.

—杭州: 浙江大学出版社, 2008.4

(高等院校计算机专业课程综合实验系列规划教材)

ISBN 978-7-308-05834-6

I. 面… II. ①胡…②翁…③陈… III. 面向对象语言—程序设计—课程设计—高等学校—教材 IV. TP312

中国版本图书馆CIP数据核字(2008)第034914号

面向对象程序设计课程设计

胡兰青 翁 恺 陈 奇 著

左保河 主审

丛书主编 何钦铭 陈根才

策 划 黄娟琴 希 言

责任编辑 黄娟琴 邹小宇

封面设计 氧化光阴

出版发行 浙江大学出版社

(杭州天目山路148号 邮政编码310028)

(E-mail: jsjcyb@zju.edu.cn)

(网址: <http://www.zjupress.com>

<http://www.press.zju.edu.cn>

<http://www.zjuxsj.com>)

电话: 0571—88925592, 88273066(传真)

排 版 浙江大学出版社电脑排版中心

印 刷 德清县第二印刷厂

开 本 787mm×1092mm 1/16

印 张 12.25

字 数 290千

版 印 次 2008年4月第1版 2008年4月第1次印刷

印 数 0001—3000

书 号 ISBN 978-7-308-05834-6

定 价 23.00元(含光盘)

版权所有 翻印必究 印装差错 负责调换

专家指导委员会

主 任

齐治昌 国防科技大学教授,教育部软件工程专业教学指导分委员会副主任

副主任

陈道蓄 南京大学计算机系教授,教育部计算机科学与技术专业教学指导分委员会副主任

蒋宗礼 北京工业大学计算机学院副院长,教授,教育部计算机科学与技术专业教学指导分委员会秘书长

委 员(按姓氏笔画排列)

王志英 国防科技大学计算机学院副院长,教授,教育部高等学校计算机科学与技术专业教学指导分委员会副主任

左保河 华南理工大学软件学院副教授,教育部高等学校软件工程专业教学指导分委员会委员

刘 强 清华大学副教授,教育部高等学校软件工程专业教学指导分委员会秘书长

孙吉贵 吉林大学计算机学院副院长,教授,教育部高等学校计算机科学与技术专业教学指导分委员会委员

庄越挺 浙江大学计算机学院副院长,教授,教育部计算机科学与技术专业教学指导分委员会委员

吴 跃 电子科技大学计算机学院教授,教育部计算机科学与技术专业教学指导分委员会委员

李 彤 云南大学软件学院院长,教授,教育部计算机科学与技术专业教学指导分委员会委员

邹逢兴 国防科学技术大学教授,国家级教学名师

陈志刚 中南大学信息学院副院长,教授,教育部高等学校计算机科学与技术专业教学指导分委员会委员

岳丽华 中国科学技术大学教授,教育部高等学校计算机科学与技术专业教学指导分委员会委员

徐宝文 东南大学教授,教育部高等学校软件工程专业教学指导分委员会委员

廖明宏 哈尔滨工业大学计算机学院副院长,教授,教育部高等学校计算机科学与技术专业教学指导分委员会委员

管会生 兰州大学信息科学与工程学院副院长,教授,教育部高等学校理工类计算机基础课程教学指导分委员会秘书长

序 言

近 10 多年来,以计算机和通信技术为代表的信息技术迅猛发展,并已深入渗透到国民经济与社会发展的各个领域。信息技术成为国家产业结构调整 and 推动国民经济与社会快速发展的最重要的支撑技术。与此同时,深入掌握计算机专业知识、具有良好系统设计与分析能力的计算机高级专业人才在社会上深受欢迎。

计算机科学与技术是一门实践性很强的学科。良好的系统设计和分析能力的培养需要通过长期、系统的训练(包括理论和实践两方面)才能获得。高等学校的实践教学一般包括课程实验、综合性设计(课程设计)、课外科技活动、社会实践、毕业设计等,基本上可以分为三个层次:第一,是紧扣课堂教学内容,以掌握和巩固课程教学内容为主的课程实验和综合性设计;第二,是以社会体验和科学研究体验为主的社会实践和课外科技活动;第三,是以综合应用专业知识和全面检验专业知识应用能力的毕业设计。课程实践(含课程实验和课程设计)是大学教育中最重要也最基础的实践环节,直接影响后继课程的学习以及后继实践的质量。由于课程设计是以培养学生的系统设计与分析能力为目标,通过团队式合作、研究式分析、工程化设计完成较大型系统或软件的设计题目的,因此课程设计不仅有利于学生巩固、提高和融合所学的专业课程知识,更重要的是能够培养学生多方面的能力,如综合能力、动手能力、文献检索能力、团队合作能力、工程化能力、研究性学习能力、创新能力等。

浙江大学计算机学院在专业课程中实施课程设计(project)已有 10 多年的历史,积累了丰富的经验和资料。为全面总结专业课程设计建设的经验,推广建设成果,我们特别组织相关课程的骨干任课教师编写了这套综合实验系列教材。本系列教材的作者们不仅具有丰富的教学和科研经验,而且是浙江大学计算机学院和软件学院的教学核心力量。这支队伍目前已经获得了两门国家精品课程以及四门省部级精品课程,出版了几十部教材。

本套教材由《C 程序设计基础课程设计》、《软件工程课程设计》、《数据结构课程设计》、《数值分析课程设计》、《编译原理课程设计》、《逻辑与计算机设计基础实验与课程设计》、《操作系统课程设计》、《数据库课程设计》、《Java 程序设计课程设计》、《面向对象程序设计课程设计》、《计算机组成课程设计》、《计算机体系结构课程设计》和《计算机图形学课程设计》等十三门课程的综合实验教材所组成。该系列教材构思新颖、案例丰富,许多案例直接取材于作者多年教学、科研以及企业工程经验的积累,适用于作为计算机以及相关专业课程设计的实验教材;也适用于对计算机有浓厚兴趣的专业人士进一步提升计算机的系统设计与分析

能力。从实践的角度出发,大部分教材配备了随书光盘,以方便读者练习。

可以说,本套教材涵盖了计算机专业绝大部分必修课程和部分选修课程,是一套比较完整的专业课程设计系列教材,也是国内第一套由研究型大学计算机学院独立组织编写的专业课程设计系列教材。鉴于书中难免存在的谬误之处,敬请读者指正,以便不断完善。

主编 何钦铭、陈根才

2007 年 6 月于求是园

前 言

以面向对象技术为基础的 C++ 程序设计方法与技术代表了目前 IT 业最先进的计算机编程方法与技术。就其影响力而言,它超过了以往任何其他程序设计语言。

学过和正在学习 C++ 程序设计方法与技术的人可能都有这样的感受:就语言的语法与使用规则,似乎比较容易理解,但要领会其广博的语法背后的语意,以及语意背后的深层思维,还是有一定难度的。很多用 C++ 语法写出来的程序还存留着明显的传统编程方法的痕迹。

为了对 C++ 语言进行有效使用,并尽快提高 C++ 程序设计方法的编程能力,编制出具有较高水平的、综合性的应用程序,学习研究并模仿若干 C++ 程序设计范例中所蕴含的程序设计技术、技巧与经验,无疑是一条迅速提高编程技术与水平的捷径。基于上述想法,浙江大学计算机学院面向对象程序设计课程组的全体骨干教师撰写了此书。这是在长期的教学、科研实践以及 ACM 大学生程序设计竞赛培训工作中总结出的教学经验,有助于读者快捷而准确地找到编程技术要领,洞穿 C++ 语法要害,直击抽象编程本质,同时有助于知识整合、问题分析与求解能力以及团队合作能力的培养。

本书是《面向对象程序设计》课程的配套教材,主要围绕 C++ 的基本知识点,设计若干个大型综合性练习案例,通过对相关背景知识回顾、题目解析与实现要点分析以及测试方法分析等,为读者完成综合性的 project 实践提供参考。

全书共分 6 章。第 1 章简要介绍了面向对象程序课程设计的组织与评分方法,课程设计实验报告的基本内容,本课程所用的工具;第 2 章以案例驱动的方式,从加深理解 C++ 语法开始,通过案例启发读者掌握四种不同(但相辅相成)的程序设计思维模式(procedural-based, object-based, object-oriented, generic programming);第 3 章以案例驱动的方式介绍容器、迭代器、算法和函数对象的使用方法;第 4 章从简单、底层开始进入 Windows 编程,从 SDK 编程到 MFC 编程,包括消息映射机制、屏幕输入/输出、Windows 应用程序的类型等;第 5 章以前面几章所给的知识点为基础,指导学生设计并实现综合性的 project;第 6 章给出部分可选题。

需指出的是:通过对本书的阅读,并不能解决所有 C++ 编程的真实问题,但作为一本《面向对象程序设计》的辅助教材,可以减少读者学习过程中的摩擦和困惑,对 C++ 的设计和实现获得深入理解。将这些理解和 C++ 标准的具体细节结合起来,就将矗立于软件开发的坚实基础之上,从而走向真正有效的 C++ 程序设计之路。

作者邮箱: lqhu@zju.edu.cn.

作 者
2007 年 9 月

目 录

第 1 章 面向对象程序设计课程设计概要	1
1.1 本课程设计的基本思路	1
1.2 实验与实践报告编写要求	1
1.2.1 实验报告的基本要求	2
1.2.2 实践报告的基本要求	2
1.3 课程设计的组织、开发与评分方法	4
1.3.1 课程设计的组织	4
1.3.2 代码与开发组织	4
1.3.3 评价与计分	4
1.4 Visual Studio .NET 集成开发环境	6
1.4.1 解决方案和项目	6
1.4.2 代码编辑、编译、生成和运行	8
1.4.3 程序调试	9
1.4.4 使用调试器	11
1.5 Cygwin/GCC	13
1.5.1 Cygwin 安装	14
1.5.2 编译、运行程序	15
第 2 章 C++ 编程基础案例分析	17
2.1 代码组织案例——设计 Complex 类	17
2.1.1 设计题目	17
2.1.2 设计分析	17
2.1.3 设计实现	27
2.2 静态多态性案例——设计 Set 类	30
2.2.1 问题引入	30
2.2.2 设计题目	30
2.2.3 设计分析	30
2.2.4 设计实现	33
2.2.5 函数重载与参数缺省值的选择	39
2.3 动态多态性案例——设计图形类层次	40

2.3.1	设计题目	40
2.3.2	设计分析	40
2.3.3	设计实现	46
2.4	文件操作案例——设计学生选课系统	50
2.4.1	设计题目	50
2.4.2	设计分析	51
2.4.3	设计实现	52
2.5	模板案例——设计模板 Array	54
2.5.1	设计题目	54
2.5.2	设计分析	54
2.5.3	设计实现	56
2.6	内存管理案例——设计 new 和 delete 的重载	57
2.6.1	问题引入	57
2.6.2	设计题目	58
2.6.3	设计分析	58
2.6.4	设计实现	59
第 3 章	STL 编程案例分析	63
3.1	容器——个人记事本案例	63
3.1.1	初涉 STL——容器	63
3.1.2	设计题目	64
3.1.3	设计分析	64
3.1.4	设计实现	66
3.2	迭代器、算法和函数对象——多项式案例	70
3.2.1	初涉 STL——迭代器、算法和函数对象	70
3.2.2	设计题目	73
3.2.3	设计分析	73
3.2.4	设计实现	75
3.3	关联容器——列车时刻查询系统案例	78
3.3.1	初涉 STL——关联容器	78
3.3.2	设计题目	78
3.3.3	设计分析	79
3.3.4	设计实现	79
第 4 章	Windows 编程案例分析	84
4.1	SDK Windows 编程——hello SDK 应用程序案例	84
4.1.1	初涉 Windows 编程	84
4.1.2	设计题目	87
4.1.3	创建步骤	87

4.1.4	设计分析	88
4.1.5	设计实现	90
4.2	MFC Windows 编程——hello MFC 应用程序案例	92
4.2.1	MFC 编程简介	92
4.2.2	设计题目	94
4.2.3	创建步骤	94
4.2.4	设计分析	95
4.2.5	设计实现	96
4.3	屏幕输出——绘图案例	98
4.3.1	图形设备接口——Windows GDI	98
4.3.2	设计题目	100
4.3.3	创建步骤	101
4.3.4	设计分析	101
4.3.5	设计实现	102
4.4	鼠标输入——井字游戏程序案例	105
4.4.1	鼠标输入	105
4.4.2	设计题目	106
4.4.3	创建步骤	106
4.4.4	设计分析	106
4.4.5	设计实现	108
4.5	键盘输入——简易编辑器案例	115
4.5.1	键盘输入	115
4.5.2	设计题目	119
4.5.3	创建步骤	119
4.5.4	设计分析	119
4.5.5	设计实现	120
4.6	基于对话框的应用程序——简易计算器案例	131
4.6.1	Windows 应用程序的类型	131
4.6.2	设计题目	131
4.6.3	创建步骤	131
4.6.4	设计分析	134
4.6.5	设计实现	135
第 5 章	综合设计案例分析	142
5.1	图书馆管理系统	142
5.1.1	系统功能描述	142
5.1.2	类体系和主要数据结构	143
5.1.3	系统功能实现	148
5.1.4	关键问题讨论	149

5.1.5	总结	157
5.1.6	完整代码概貌	157
5.1.7	练习	158
5.2	五子棋游戏	158
5.2.1	系统功能描述	158
5.2.2	主要数据结构	158
5.2.3	主要算法	160
5.2.4	关键技术讨论	160
5.2.5	创建步骤	162
5.2.6	完整代码概貌	162
5.2.7	练习	163
5.3	MiniCAD	163
5.3.1	系统功能描述	163
5.3.2	类体系和主要数据结构	164
5.3.3	系统功能实现	164
5.3.4	完整代码概貌	171
第6章	综合设计习题	172
6.1	加强版编辑器	172
6.1.1	问题描述	172
6.1.2	解题思路	172
6.1.3	关键提示	173
6.2	通讯录管理软件	174
6.2.1	问题描述	174
6.2.2	解题思路	174
6.2.3	关键提示	174
6.3	打字练习软件	178
6.3.1	问题描述	178
6.3.2	解题思路	178
6.3.3	关键提示	178
6.4	即时通讯系统	178
6.4.1	问题描述	178
6.4.2	解题思路	178
6.4.3	关键提示	178
6.5	邮件系统	179
6.5.1	问题描述	179
6.5.2	解题思路	179
6.5.3	关键提示	179
6.6	版本管理软件	179

6.6.1	问题描述.....	179
6.6.2	解题思路.....	180
6.6.3	关键提示.....	181

第 1 章

面向对象程序设计课程设计概要

1.1 本课程设计的基本思路

面向对象程序设计是当今软件设计开发的基本思想、方法与技术。本书的目的是与面向对象程序设计课程教学相配合,强化读者面向对象程序设计的实践训练,从而领悟面向对象方法与技术的精髓。

程序设计是一门实践性很强的课程,它还是一门艺术,没有标准答案。从软件开发目标的确定、对象数据和功能的抽象、程序代码的具体组织,到一些很具体的代码编写技巧,程序设计是一个极具创造性的过程。在设计开发的路上,有多种选择,因而课程设计自始至终都需要编程者大胆创新。

编程没有标准答案,但程序需要追求美。尽管程序的“美”没有标准答案,但仍然有一个大多数一流软件人员认可的评判框架。为写美的程序改进支撑技术,是软件界多年来的一个努力方向。具体到一个实际的程序,美也是一个可以不断改进的追求。正是基于这样的理解,本书进行了大量案例的“讨论”,旨在传达这样一个理念,编程是一个需要大家共同感悟的美妙过程。

面向对象编程思想与 C++ 的语法内容是相当庞大的一套体系。本书不是面面俱到地讲授 C++ 的具体内容,而是通过案例来讨论 C++ 及面向对象编程中的一些要点。案例是渐进式展开的,因此接近于实际编程中的逐步求精和迭代过程,目的是使读者感受到面向对象编程技术的精彩之处。

一些大的综合课程设计项目讨论是本书的一个重要内容,但它们不是可供套用的“范本”,而是全面地展示一些肯定充满瑕疵但尽量接近真实的“过程”。本书第 6 章的综合设计习题是建议性的,我们鼓励读者自己改进或重新设计题目。

1.2 实验与实践报告编写要求

文档编写是软件开发的重要组成部分,本课程设计希望通过实践(实验)报告的编写,培养学生具备基本的分析、总结与文档表述的能力。

本课程设计规范了对实践(实验)报告的编写。这里实验报告与实践报告的区别在于其综合性与完整性。一般来说,个人完成、规模不大、主要是练习语法与算法的,采用实验报

告;多个人协作完成,结果是一个完整的系统的,采用实践报告。

1.2.1 实验报告的基本要求

实验报告一般针对个人独立完成,规模在 300~500 行代码以内的程序。实验报告由以下部分组成。

1. 实验目标

这一部分内容说明本实验希望训练学生哪些能力、掌握哪些知识,以及实验程序需要完成的功能目标。

训练目的可以简略说明,如培养学生运用继承性与虚函数进行面向对象抽象编程的能力。

实验程序的功能目标需要详细说明,完整列出程序的主要功能,同时对这些功能的性能要求进行必要的讨论。

2. 基本实现思路

这一部分内容对程序的主要类体系进行完整的说明,对其中涉及的主要数据组织与算法进行说明。

3. 关键技术与难点讨论

这一部分内容对整个程序涉及的关键技术与难点进行深入的分析、研究与讨论。包括程序实现时不同数据结构的比较与选择、不同算法的比较与选择、主要实现难点及对策。这里的实现难点包括程序开发过程中走的弯路以及克服经过的讨论。

4. 对程序结果的分析与改进的展望

这一部分内容包括程序测试的分析与讨论,说明程序设计的测试方案以及测试的结果。另外,对程序的稳定性、鲁棒性、正确性、运行效率、交互友好性等进行全面的分析与讨论;对程序存在的问题进行全面、准确和客观的说明。最后,对程序的改进方案进行详细的讨论。

5. 附录

附录中应包含源代码,要求程序风格清晰、易理解,有充分的注释。

1.2.2 实践报告的基本要求

实践报告一般针对由一个编程小组完成、规模在 1000 行代码以上的软件系统。实践报告由以下部分组成。

1. 实践目标

这一部分内容说明本实践希望达到哪些综合训练目的,以及软件需要完成的功能目标。综合训练目的可以简略说明,如训练多层次组织软件系统、多层次分工组织开发团队。软件的功能目标需要详细说明,主要包括基本功能目标、扩展功能目标与高级功能目

标。基本功能目标是软件必须完成本实践任务最低要求的功能目标,这是软件迭代开发第一轮应该完成的功能。扩展功能目标是指软件推荐完成的功能目标,一般指软件迭代开发中第二轮完成的功能。高级功能目标是指实践可以努力去完成的目标,包括学生自己设想出来的功能目标,一般是在软件迭代开发以后完成。

组长需对这一部分内容进行完整的、详细的说明,其他成员只需对与自己分工关系比较密切的功能进行详细说明。对于详细说明的功能,同时要求对性能要求进行必要的讨论。

2. 基本实现思路

这一部分内容对软件的总体架构进行说明,包括对不同层次子系统的主要类体系进行完整的说明,对其中涉及的主要数据组织与算法进行说明。

除了最基本的部分外,组内成员要详细讨论各自承担的那部分工作的技术实现思路。

3. 关键技术与难点讨论

这一部分内容对所涉及的关键技术与难点进行深入的分析、研究与讨论。包括对软件实现不同架构的比较与选择、实现过程中对不同数据结构的比较与选择、不同算法的比较与选择、主要实现难点及对策。这里的实现难点包括程序开发过程中走的弯路以及如何克服的经过的讨论。

4. 对软件结果的分析与改进

这一部分内容包括对软件测试的分析与讨论,说明软件设计的测试方案以及测试的结果。各组员从自己负责的那部分工作的角度,对软件的稳定性、鲁棒性、正确性、运行效率、交互友好性等进行全面的分析并展开讨论,对软件中与自己设计开发有关的问题进行全面、准确和客观的说明。另外,各组员应对软件中与自己有关的部分的改进方案进行详细的讨论。

5. 软件开发组织与管理分析

这一部分内容,各组员应对自己进行设计开发的时间进度、工作量进行说明与分析,对项目的分工与组织进行分析与说明,另外应说明自己与相关人员的配合协同情况。

6. 附录

附录中应包含源程序、软件操作手册、软件用户手册、软件工程文件或 Makefile 文件。

7. 任务分配

- 程序员:XXX
- 测试员:XXX
- 文档员:XXX

提交日期:yyyy-mm-dd

1.3 课程设计的组织、开发与评分方法

这里首先需要强调的是,本课程设计主要是训练学生的面向对象编程技术能力,不模拟软件工程中严格的分工协作。

本课程设计的题目分为两类,一类是进行针对性训练的题目,一类是进行综合性训练的题目。前者要求一个人完成,后者要求分组完成。

1.3.1 课程设计的组织

一个人完成的题目比较简单,由教师布置题目,学生按要求完成后提交实验报告和相关附件。

程序小组完成的综合性实践比较复杂。先由学生以自荐或推荐的方式给出组长候选名单,由实验教师根据前面单人实验的情况给出组长名单,由组长招募组员。每组一般为4人(包括组长),个别情况可以为3人或5人。

软件设计由开发小组所有成员讨论提出,组长有责任做最后决定。程序总体架构、分工划分与接口设计完成后,各位组员都需要承担一定的详细设计、实现与单元测试工作,最后的集成测试由组长协调全组人员进行。

实验报告要求每位同学分别编写。

1.3.2 代码与开发组织

从系统总体架构设计开发,要充分利用面向对象的方法完成良好的代码组织。软件技术发展的一条基本脉络就是软件划分技术的发展,从函数模块到类再到组件化,等等,无不是软件划分技术的重要里程碑。

课程设计一开始,就应该确定软件的总体架构、代码的划分方案,由此确定开发的分工方案;可以是分块开发,如一个进销存系统,由不同的人分别写采购、库存和销售;也可以是分层组织,如一个人写持久化存储部分,一个人写内存数据组织与处理部分,一个人写前端表现部分等等;当然还可以是混合的划分方式。不管采用哪种分工方式,一定要为每个组员分配合理的程序开发任务、合理的测试任务(很多时候,测试可以交换进行,如张三开发的程序由李四进行测试,而李四开发的程序交由张三进行测试)。

1.3.3 评价与计分

1. 专项实验评价

计分由程序计分与实验报告计分两部分组成。程序部分占60%,实验报告占40%。

(1)程序60%的计分,包括以下几部分:

源代码的规范性(10%):代码是否有良好的注释,类、函数与变量取名是否易于理解,源代码的阶梯形结构是否清晰,程序段落是否有合理的空行,等等。

程序的功能(30%):是否覆盖了全部功能点,每一项功能实现是否充分。

程序的可扩展性(5%):程序从类体系到数据结构的设计,是否考虑了今后功能与性能(如数据处理能力)等的扩展。

程序的鲁棒性(5%):执行程序在测试时是否发生意外中止等情况。

程序的正确性(5%):程序是否在各种测试集下都能得到正确结果。

程序的交互友好性(5%):用户是否能方便地使用,高效地交互。

(2)实验报告 40%的计分,包括:

报告的格式(4%)。

功能分析(5%)。

主要技术思路(8%)。

关键技术分析(10%)。

结果分析(8%)。

展望(5%)。

2. 综合实践评价

计分由软件计分和实践报告计分两部分组成。软件占 60%,实践报告占 40%。

(1)软件评价以整个软件进行评价,再根据组内成员的贡献分到组员。软件 60%的计分,包括:

源代码的规范性与一致性(10%):代码是否有良好的注释,类、函数与变量取名是否易于理解,源代码的阶梯形结构是否清晰,程序段落是否有合理的空行,不同组员开发的程序风格是否统一,等等。

软件的功能(30%):覆盖了哪些功能点,每一项功能实现是否充分。基本功能、扩展功能与高级功能设计是否合理。

软件的可扩展性(5%):软件从类体系到数据结构的设计是否考虑了今后功能与性能(如数据处理能力)等的扩展。

软件的鲁棒性(5%):执行程序在测试时是否发生意外中止等情况。

软件的正确性(5%):软件是否在各种测试集下都能得到正确结果。

软件的交互友好性(5%):用户是否能方便地使用、高效地交互。

(2)实验报告 40%的计分,包括:

报告的格式(3%)。

功能分析(5%)。

主要技术思路(7%)。

关键技术分析(10%)。

结果分析(5%)。

展望(5%)。

软件开发组织与管理(5%)。