

<<软件工程实践教学>>

图书基本信息

书名：<<软件工程实践教学>>

13位ISBN编号：9787111377597

10位ISBN编号：7111377591

出版时间：2012-5

出版时间：机械工业出版社

作者：刘冰 等编著

页数：399

字数：636000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件工程实践教学>>

内容概要

《软件工程实践教学(第2版)》详细介绍了软件工程、软件开发过程、软件计划、需求分析、总体设计、详细设计、编码、软件测试、软件维护、软件工程标准化和软件文档、软件工程质量、软件工程项目管理以及软件工程开发实例。

各章均配有习题，部分章后附有经典例题讲解和实验内容。

《软件工程实践教学(第2版)》可作为高等学校计算机专业课程的教材或教学参考书，也可作为通信、电子信息、自动化等相关专业的计算机课程教材，还可供软件工程师、软件项目管理者 and 应用软件开发人员阅读参考。

本书由刘冰、刘锐、瞿中、吴渝编著。

<<软件工程实践教学教程>>

书籍目录

出版说明

前言

第1章 软件工程概述

1.1 软件

1.1.1 软件的特点

1.1.2 软件的发展

1.1.3 软件危机

1.1.4 软件工程的定义

1.1.5 软件工程的三要素

1.1.6 软件工程方法

1.1.7 软件工程的发展历史

1.2 软件过程的概念

1.2.1 软件生命周期及软件开发过程的定义

1.2.2 软件开发过程的制品

1.2.3 软件开发工具

1.3 软件过程模型

1.3.1 瀑布模型

1.3.2 快速原型模型

1.3.3 增量模型

1.3.4 螺旋模型

1.3.5 喷泉模型

1.3.6 形式化方法模型

1.3.7 基于组件的开发模型

1.4 Rational统一开发过程

1.5 实例：软件外包开发过程

1.5.1 外包软件的定义

1.5.2 外包软件的开发特点

1.5.3 改进方案

1.5.4 实施、总结

1.6 小结

1.7 习题

第2章 软件开发方法简介

2.1 面向结构的软件工程

2.2 结构化方法遵循的基本原则及核心问题

2.3 结构化方法的特点

2.4 基于Microsoft Office Visio 2007的结构化建模

2.4.1 Microsoft Office Visio 2007简介

2.4.2 Microsoft Office Visio 2007工作环境

2.4.3 Microsoft Office Visio 2007操作入门

2.4.4 实践案例

2.5 面向对象的软件工程

2.6 面向对象方法的基本原则和核心问题

2.7 面向对象方法的特点与优势

2.8 面向对象的基本概念

2.9 面向对象方法学的研究及实践领域

<<软件工程实践教学教程>>

2.10 面向对象的统一建模语言介绍

- 2.10.1 统一建模语言概述
- 2.10.2 UML的基本实体
- 2.10.3 常用的UML图

2.11 基于Rational Rose 2003的UML建模

- 2.11.1 Rational Rose 2003简介
- 2.11.2 Rational Rose 2003建模环境
- 2.11.3 Rational Rose 2003模型的视图
- 2.11.4 Rational Rose 2003建模界面

2.12 结构化方法与面向对象方法比较

- 2.13 小结
- 2.14 习题

第3章 结构化分析

3.1 可行性研究

- 3.1.1 问题定义
- 3.1.2 可行性研究的内容
- 3.1.3 可行性研究的步骤

3.2 需求分析

3.3 获取需求的方法

3.4 结构化分析方法

- 3.4.1 数据流图
- 3.4.2 数据字典
- 3.4.3 实体关系图
- 3.4.4 状态转换图

3.5 快速原型分析方法

3.6 结构化分析实例

3.7 小结

.....

第4章 结构化设计

第5章 结构化的实现

第6章 面向对象分析方法

第7章 面向对象的设计

第8章 面向对象的实现及测试

第9章 运行和维护

第10章 软件工程标准化和软件质量

第11章 软件工程项目管理

第12章 结构化开发实例

第13章 面向对象软件开发实例

参考文献

章节摘录

版权页：插图：7.灰盒测试的优点（1）能有效地发现黑盒测试的盲点 通过了解代码的内部实现，补充功能测试用例。

这需要灰盒测试人员在查看代码之前，掌握需求，并清楚已有的功能测试用例。

对某一功能点的实现进行代码分析（白盒测试），然后与黑盒测试（功能测试）充分结合起来，相互弥补。

（2）可以避免过度测试，精简冗余用例 例如，具有相同特点的某一功能，在进行功能测试时，是否每个界面或提示框、对话框都需进行该功能的测试。

在没有采用灰盒测试之前，确实进行过这样的穷举测试，但收效甚微。

（3）能及时发现没有来源的更改 特别是在产品的维护阶段，每一行代码的更改都必须要有更改来源，但实际开发过程中，并不是每个开发人员都能做到的，也并不是每个人都能清楚意识到更改后没有得到有效验证所带来的后果。

所以测试人员采取事后代码审查的方法也是不得已而为之的事。

【例】PDA闹钟事件的测试 背景描述：闹钟是PDA（Personal Digital Assistant）上的一个应用程序，它的主要功能就是在用户设置的时间点到达时，进行响铃提醒。

响铃时，无论用户当前在做什么操作，会弹出一个响铃提示界面。

提出问题：一天，负责测试此模块的工程师小叶提出一个问题“闹铃事件优先级高，在所有应用程序界面、对话框或提示框上面都会弹出，黑盒测试需要进入所有情况的用户界面，然后等待闹铃事件的发生，才能全面验证到各种情况的用户使用场景。

”但这样的穷举测试，耗时太多，效果也并不一定好。

分析问题：关于闹铃的应用，实际上该模块只提供了一个接口。

如能从代码角度分析到所有其他模块调用的是同一个闹铃接口函数（对于响铃及停止响铃后对原界面的恢复，都是由一个统一接口处理的），那么再通过黑盒功能测试，在某一个用户界面进行此响铃功能的验证，即可证明代码的实现是符合需求的，这正是灰盒测试的方法。

解决问题：后来，对于闹铃功能点的系统验证，采用了灰盒测试的方法，做了代码正向、逆向分析，结合功能性用户场景进行验证，节省了近2/3的测试时间。

5.5.4人工测试 人工测试指的是采用人工方式进行测试，目的是通过对程序静态结构的检查，找出编译时不能发现的错误。

经验表明，组织良好的人工测试可以发现程序中30%~70%的编码和逻辑设计错误。

人工测试又称为代码审查，其内容包括检查代码和设计是否一致，检查代码逻辑表达是否正确和完整，检查代码结构是否合理等。

主要有3种方法。

1) 个人复查：指程序员本人对程序进行检查。

由于心理上的原因和思维惯性的影响，自己的错误一般不容易发现，对功能理解的错误更不可能纠正。

因此，这种方法主要针对小规模程序，效率不高。

<<软件工程实践教学>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>