

<<实用软件工程学>>

图书基本信息

书名：<<实用软件工程学>>

13位ISBN编号：9787508464398

10位ISBN编号：7508464397

出版时间：2009-6

出版时间：水利水电出版社

作者：庄晋林，杨志宏 主编

页数：342

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>

## &lt;&lt;实用软件工程学&gt;&gt;

## 前言

在信息技术广泛普及的今天，作为信息化重要基础的软件产业已经成为国际竞争的焦点和各国竞相发展的战略性先导产业。

国家先后出台了一系列政策、法规推动和鼓励软件产业的发展。

软件产业是指从事软件开发、销售及相关服务等业务的企业组成的集合体。

其中软件开发的技术水平是决定整个软件产业发展的关键因素。

软件工程学是计算机科学与技术学科中研究软件开发的一个重要分支学科。

在20世纪60年代，伴随着西方国家现代化进程的不断加快，计算机软件技术在各行各业迅速普及，人们对软件的需求日益复杂化，对软件质量的要求也越来越高，软件开发面临着日益严峻的挑战。

在“软件危机”的强烈冲击下，软件工程学应运而生，经过40余年的发展，如今已经成为包含一整套的理论、方法、技术和工具的较为完善的学科体系。

但是，正如著名软件工程学者B. W. Boehm总结软件工程的基本原理时提出的，要“承认不断改进软件工程实践的必要性”，从而表明在当今软件工程技术日趋成熟的条件下，仍然需要不断改进、发展，朝着最终彻底消除“软件危机”的目标前进。

本书的适用范围主要是作为计算机科学与技术及相关专业应用型本科或专科学生的教材或教学参考书。

书中从实用的角度，力求点面兼顾、深入浅出地介绍软件工程学的基本概念、方法和技术。

希望读者通过阅读和学习，能够了解软件开发过程中的主要活动，初步认识当今主流的软件开发方法及其关键技术。

在深入理解基本概念、基本原理的同时，动手实践是掌握软件工程方法和技术的唯一途径。

为此，在部分章节中给出了应用案例，可用于教学中案例分析的素材。

并在附录中给出了课程设计内容，读者可以通过多人合作参与课程设计或小型项目的开发，逐步培养团队精神和工程化意识，为今后从事软件开发工作奠定良好的基础。

## <<实用软件工程学>>

### 内容概要

《实用软件工程学》从实用的角度，力求点面兼顾、深入浅出地介绍软件工程学的基本概念、方法和技术。

同时，在注重基本概念、基本原理的基础上，强调了实践的重要性和必要性，部分章节给出了应用案例，可作为案例教学的素材，并在附录中编排了课程设计内容，供读者进行实践教学。

全书分为10章，主要内容包括软件工程学概述、软件需求分析、软件设计（包括概要设计与详细设计）、编码与语言选择、软件测试、面向对象方法、软件复用、软件维护、软件项目管理与软件质量保证简介、软件工程环境。

《实用软件工程学》的适用范围主要是作为计算机科学与技术及相关专业应用型本科层次或专科学生的教材，也可作为相关专业的教师或工程技术人员的参考书。

## &lt;&lt;实用软件工程学&gt;&gt;

## 书籍目录

前言第1章 软件工程学概述 1.1 软件发展史与软件危机 1.2 软件工程学的概念 1.3 软件生命周期 1.4 软件过程模型 1.5 传统软件工程与面向对象软件工程 1.6 软件开发工具 1.7 小结 习题1 参考文献第2章 软件需求分析 2.1 软件需求分析的前期工作 2.2 软件需求分析的重要性 2.3 软件需求获取的常用方法 2.4 分析建模 2.5 结构化分析方法 2.6 其他图形工具 2.7 软件需求规格说明和需求验证 2.8 应用案例 2.9 小结 习题2 参考文献第3章 软件设计 3.1 软件设计概述 3.2 软件概要设计 3.3 软件设计的基本原理 3.4 软件设计的准则 3.5 用户界面设计 3.6 概要设计工具 3.7 结构化设计方法——面向数据流的设计方法 3.8 详细设计 3.9 设计文档及其复审 3.10 小结 习题3 参考文献第4章 编码与语言选择 4.1 编码语言的选择 4.2 编码的风格 4.3 程序设计方法 4.4 小结 习题4 参考文献第5章 软件测试 5.1 软件测试概述 5.2 黑盒测试 5.3 白盒测试 5.4 软件测试的过程 5.5 软件纠错 5.6 小结 习题5 参考文献第6章 软件维护 6.1 软件维护的定义和分类 6.2 软件维护的特点 6.3 软件可维护性 6.4 维护过程与维护活动 6.5 软件维护的副作用 6.6 软件再工程 6.7 小结 习题6 参考文献第7章 面向对象方法 7.1 面向对象方法概述 7.2 面向对象的分析 7.3 面向对象的设计 .....第8章 软件复用第9章 软件项目管理与软件质量保证简介第10章 软件工程环境

## 章节摘录

插图：其缺点在于需要开发和维护大量的桩模块。

而桩模块并不能够反映实际情况，涉及复杂算法和真正输入/输出的模块一般在底层，这样重要数据不能及时回送到上层模块，因而测试并不充分和完善，一旦发现问题，将导致大量的回归测试。

所以这种方法有它的局限性，如果不能使桩模块正确地向上传递有用的信息，测试人员可选择以下几种解决方法：1) 把很多测试推迟到用真实模块代替了桩模块之后进行，但这将使我们在确定错误原因时比较困难。

2) 进一步开发能模拟实际模块的桩模块，但这样会大大地增加开销。

3) 从层次结构的底部向上组装和测试软件。

此种方法较切实可行，下面将详细介绍。

### (2) 自底向上集成测试。

自底向上集成测试是从软件结构最低层的模块开始，从低到高对模块进行组装和测试。

因为是从底部向上结合模块的，对于一个给定层次的模块，它的子模块及其子模块的所有下属模块已经组装并测试完成，所以不再需要桩模块。

在模块的测试过程中需要从子模块得到的信息可以直接运行该子模块得到。

自底向上集成测试的步骤如下：第一步，为最底层模块编写驱动模块，对最底层模块进行测试，最底层模块之间的测试可以并行进行。

也可以把低层模块组合成实现某一特定软件子功能的族，为其编写驱动模块，控制它进行测试。

第二步，用实际模块代替驱动模块，与它已测试过的直属子模块组装成为一个子系统。

第三步，为新形成的子系统开发驱动模块，对其进行测试。

第四步，若该子系统已成为主控模块，即最高层模块，则结束集成；否则转第二步。

以图5-15所示的软件结构为例，用图5-17所示说明自底向上组装和测试的过程。

## <<实用软件工程学>>

### 编辑推荐

《实用软件工程学》由中国水利水电出版社出版。

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>