

<<编程原本>>

图书基本信息

书名：<<编程原本>>

13位ISBN编号：9787111367291

10位ISBN编号：7111367294

出版时间：2012-1-10

出版时间：机械工业出版社华章公司

作者：Alexander Stepanov,Paul McJones

页数：296

译者：裘宗燕

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>

前言

前言： 本书将演绎方法应用于程序设计，讨论程序与保证它们能正确工作的抽象数学理论之间的联系。书中把反映这些理论的规格（specification），基于这些理论写出的算法，以及描述算法性质的引理和定理一起呈现给读者。这些算法在一种实际程序设计语言里的实现是本书的中心。虽然规格主要是供人阅读，但它们也应该（或者说必须）严格地与非形式化的、供机器使用的代码相结合，必须在通用的同时又是抽象而且精确的。

与在其他科学和工程领域里的情况一样，适合作为程序设计的基础的同样是演绎方法。演绎方法能帮助我们将复杂系统分解为一些具有特定数学行为的组件，而这种分解又是设计高效、可靠、安全和经济的软件的前提。

本书是想奉献给那些希望更深入地理解程序设计的人们，无论他们是专职软件开发人员，还是把程序设计看作其专业活动中一个重要组成部分的科学家或工程师。

本书编写的基本想法是让读者从头到尾完整阅读。

读者只有通过阅读代码、证明引理、完成练习，才能真正理解书中的各方面材料。

此外我们还建议了一些项目，其中有些是完全开放的。

本书的内容很紧凑，认真的读者最终会看到书中各部分之间的联系，以及我们选择这些材料的理由。发现本书在体系结构方面的原理应该是读者的一个目标。

我们假定读者已经具有完成各种基本代数操作的能力。

还假定读者熟悉逻辑和集合论的基本术语，如普通本科生在离散数学课程中学习的内容。

附录A总结了书中使用的各种记法。

如果在一些特定的算法里需要某些抽象代数的概念，书中会给出相应的定义。

我们还假定读者熟悉程序设计，理解计算机体系结构，理解最基本的算法和数据结构。

我们选用C++，是因为它组合了强有力的抽象设施和基础机器的正确表示。

这里只用了该语言的一个小子集，需求被写成程序里的结构化注释。

我们希望不熟悉C++的读者也能阅读本书。

附录B描述了书中使用的C++子集。

在书中的任何地方，在需要区分数学记法和C++的地方，根据所用的字体、排版和上下文就能确定用的是哪种意义（是数学的还是C++的）。

虽然书中的许多概念和程序与STL（C++标准模板库）里的东西对应，但这里的一些设计决策是与STL不同的。

书中还忽略了实际程序库（如STL）必须考虑和处理的许多问题，如名字空间、可见性、inline指令等等。

第1章描述值、对象、类型、过程和概念。

第2~5章描述各种代数结构（如半群、全序集）上的算法。

第6~11章讨论抽象内存上的算法。

第12章讨论包含对象成员的对象。

跋给出了我们对本书中阐释的工作途径的反思。

<<编程原本>>

内容概要

本书提供了有关编程的一种与众不同的理解。其主旨是，实际的编程也应像其他科学和工程领域一样基于坚实的数学基础。本书展示了在实际编程语言（如C++）中实现的算法如何在最一般的数学背景中操作。例如，如何定义快速求幂算法，使之能使用任何可交换运算。使用抽象算法将能得到更高效、可靠、安全和经济的软件。

这不是一本很容易读的书，它也不是能提升你的编程技能的秘诀和技巧汇编。本书的价值是更根本性的，其终极目标是提升你对编程的洞察力。要想从中大获裨益，你需要从头到尾认真学习：阅读代码，证明引理，完成练习。到结束之时，你将看到如何把这里讨论的演绎方法应用到你的程序中，保证你做出的软件部件能一起工作，并表现出它们所应该表现的行为。

书中给出的算法和需求针对某些被操作的类型。有关这些描述的代码（也可以通过Web得到）采用C++的一个小子集书写，这样做是为了让所有有经验的程序员都能理解。这个小子集可以看做一种特殊语言，是由Sean Parent和Bjarne Stroustrup一起设计的。

无论你是一位软件开发者，还是其他以编程作为一项重要活动的专业人员，或者是一名在校的学生，你都会逐渐理解本书的经验丰富的作者多年来一直在教授和阐释的道理：数学对于编程是绝好的东西，理论对于实际是绝好的东西。

作者简介

Alexander Stepanov

于1967~1972年间在国立莫斯科大学学习数学，从1972年开始在苏联，1977年移民后继续在美国从事编程工作。

他编写过操作系统、编程工具、编译器和各种程序库。

他在程序设计基础方面的工作先后得到GE、Polytechnic、AT&T、惠普、SGI和Adobe的支持。

1995年因C++标准模板库的设计获Dr.

Dobb, Journal的程序设计杰出贡献奖。

Paul McJones

于1967~1971年间在加州大学伯克利分校学习工程数学。

从1967年开始介入程序设计，涉足的领域包括操作系统、程序设计环境、事务处理系统以及企业和客户应用系统等。

他先后在加州大学、IBM、Xerox、Tandem、DEC和Adobe工作。

1982年他与合作者一起因论文“ The

Recovery Manager of the System R Database

Manager ” 获得ACM程序设计系统和语言论文奖。

裘宗燕，北京大学数学学院信息科学系教授。

长期从事计算机软件理论，形式化方法，程序设计和程序设计语言等方面的教学和研究工作。

译著包括《程序设计实践》、《从规范出发的程序设计》、《C++语言的设计和演化》、《C++程序设计语言》、《计算机程序的构造和解释》、《程序设计语言——实践之路》等，著作有《从问题到程序——程序设计与C语言引论》等。

<<编程原本>>

书籍目录

译者序

前言

关于作者

第1章 基础

1.1 理念范畴: 实体, 类别, 类属

1.2 值

1.3 对象

1.4 过程

1.5 规范类型

1.6 规范过程

1.7 概念

1.8 总结

第2章 变换及其轨道

2.1 变换

2.2 轨道

2.3 碰撞点

2.4 轨道规模的度量

2.5 动作

2.6 总结

第3章 可结合运算

3.1 可结合性

3.2 计算乘幂

3.3 程序变换

3.4 处理特殊情况的过程

3.5 参数化算法

3.6 线性递归

3.7 累积过程

3.8 总结

第4章 线性序

4.1 关系的分类

4.2 全序和弱序

4.3 按序选取

4.4 自然全序

4.5 派生过程组

4.6 按序选取过程的扩展

4.7 总结

第5章 有序代数结

5.1 基本代数结构

5.2 有序代数结构

5.3 求余

5.4 最大公因子

5.5 广义gcd

5.6 Steingcd

5.7 商

5.8 负量的商和余数

<<编程原本>>

- 5.9概念及其模型
- 5.10计算机整数类型
- 5.11结论
- 第6章 迭代器
 - 6.1可读性
 - 6.2迭代器
 - 6.3范围
 - 6.4可读范围
 - 6.5递增的范围
 - 6.6前向迭代器
 - 6.7索引迭代器
 - 6.8双向迭代器
 - 6.9随机访问迭代器
 - 6.10总结
- 第7章坐标结构
 - 7.1二叉坐标
 - 7.2双向二叉坐标
 - 7.3坐标结构
 - 7.4同构,等价和有序
 - 7.5总结
- 第8章 后继可变的坐标
 - 8.1链接迭代器
 - 8.2链接重整
 - 8.3链接重整的应用
 - 8.4链接的二叉坐标
 - 8.5结论
- 第9章拷贝
 - 9.1可写性
 - 9.2基于位置的拷贝
 - 9.3基于谓词的拷贝
 - 9.4范围的交换
 - 9.5总结
- 第10章 重整
 - 10.1置换
 - 10.2重整
 - 10.3 反转算法
 - 10.4 轮换算法
 - 10.5 算法选择
 - 10.6 总结
- 第11章 划分和归并
 - 11.1划分
 - 11.2平衡的归约
 - 11.3归并
 - 11.4总结
- 第12章 复合对象
 - 12.1简单复合对象

<<编程原本>>

12.2动态序列

12.3基础类型

12.4总结

跋

附录A 数学表示

附录B 程序设计语言

参考文献

索引

<<编程原本>>

章节摘录

版权页：插图：

<<编程原本>>

媒体关注与评论

“要是问一位机械、建筑或电子工程师,如果不依靠坚实的数学基础,他们能走多远。

他们会告诉你‘走不了多远’。

而所谓的软件工程师在实践其技能时,却常常对他们所做工作的数学基础知之甚少,甚至一无所知。

同时我们也很奇怪为什么软件由于不能按时发布并充斥错误而声名狼藉,而其他工程师却能按时完成其桥梁、汽车、各种电子装置等,而且缺陷很少。

本书就是想纠正这种不平衡现象。

我在Adobe的高级开发团队的成员们,但凡参加了基于同样材料的课程,都觉得付出的时间获益匪浅。

初看可能觉得这种高度技术性的文字只是为计算机科学家写的,其实所有从事实际工作的软件工程师都应该来读。

” —— Martin Newell, Adobe 院士 “ 本书包含一些我所见过的最美的代码。

” —— Bjarne Stroustrup, C++ 设计者 “ 我很高兴看到Alex课程的内容。

担任Silicon Graphics的CTO时,我曾大力支持这一课程的开发和教授,现在这本书已经能被所有程序员阅读了。

” —— Forest Baskett, 合伙人, New Enterprise Associates “ Paul的耐心和在体系结构方面的经验帮助把Alex的数学方法组织成为一套高度结构化的大厦——功德无量!” —— Robert W. Taylor, Xerox PARC CSL和DEC系统研究中心创始人

<<编程原本>>

编辑推荐

《编程原本》编辑推荐：追溯数学原理,探求编程的本质，STL之父Alexander Stepanov力作，C++之父Bjarne Stroustrup鼎力推荐，北大数学学院教授裘宗燕老师倾情献译。

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>