

<<系统程序员成长计划>>

图书基本信息

书名：<<系统程序员成长计划>>

13位ISBN编号：9787115224019

10位ISBN编号：7115224013

出版时间：2010-04

出版时间：人民邮电出版社

作者：李先静

页数：272

字数：413000

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>

<<系统程序员成长计划>>

前言

在经历过几个大型的、失败的项目之后，我终于认识到了：没有什么比高素质的程序员更能决定项目的成功。

无论采用什么开发过程，什么编程语言和开发工具，离开了高素质的程序员，都是白费力气。

毫无疑问，人是软件开发中最重要的因素。

但并非每个人都重要，也不是什么样的人重要，在软件开发中，只有那些高素质的程序员和那些对项目有突出贡献的人才是重要的。

不过高素质的程序员并不多见，所以从我开始带人起，就一直在思考团队成员培养的问题。

我做过很多尝试，从小组内学习到整个部门一起上大课，最后又回到对个人做单独的辅导；从通过Code Review（代码评审）做现场教育到制定一个宏伟的培训计划，最后又回到一个朴素的培训过程。

其中遇到了很多问题，开始是培训不够系统，效果不甚理想，后来又因为计划过于“宏伟”而无法实施，等到最后形成一个朴素的、切实可行的培训方案，已经经过了好几年时间，直到去年，整个计划才趋于完善。

我把这个培训计划称为系统程序员成长计划，而这正是我在本书中所要介绍的。

培训内容不是来源于某本书，毕业八年来，我坚持不懈地阅读有关书籍，所读过的300多本不同类型的著作装满了家中的7个大储物箱，而这些著作囊括了大部分经典的IT图书。

当然培训的内容也不是全部源于书本，这几年我在开发开源软件的过程中所收获的感悟和积累的经验也融入其中。

我的培训计划并不是要阐述什么高深的道理，相反，我这本书主要是针对应届毕业生和业余爱好者写的，目的就是为了让初学者进阶为一个专业的程序员。

为什么把这个培训计划叫做“系统程序员成长计划”，而不是“程序员成长计划”呢？

程序员的范围太广了，虽然软件开发有很多相似之处，但是隔行如隔山，比如对于目前炙手可热的Web开发，我完全是外行。

为了不造成“想什么都讲一点，结果是什么都没有讲清楚”的尴尬，我得把培训计划限定在我熟悉的范围之内。

而所谓系统程序员，是指从事操作系统内核、DBMS、Gui系统、基础函数库、应用程序框架、编译器和虚拟机等基础软件开发的程序员。

<<系统程序员成长计划>>

内容概要

本书以生动的语言和丰富的代码示例，运用一些相对简单的例子分析开发系统程序中可能遇到的各种问题。

作者把数年的开发经验和阅读大量书籍的体会，结合他在培训新员工过程中所积累的培养方法，融会贯通在这12章的内容中。

书中介绍了链表、数组、栈、队列和散列表等基础数据结构，也介绍了并发、同步和内存管理等系统程序中常需注意的问题，还讲解了文本处理器等具体应用程序的设计方法。

本书是初涉系统程序开发领域的人不可多得的一本参考书。

书中体现的思想对于其他各种软件开发人员、相关专业的在校学生以及软件开发爱好者都有启发意义。

。

<<系统程序员成长计划>>

作者简介

李先静，CSDN开源专家，有着十年Linux开发经验、五年手机开发经验。

擅长嵌入式程序员培训，软件架构设计和技术写作。

近几年负责Broncho Linux智能手机项目，致力于基于Linux的嵌入式系统的学习和研究。

其CSDN博客连续三年被CSDN提名为最有价值的技术博客（MVB），他先后发表了近500篇技术类博文，博客文章被各大技术网站转载。

在《程序员》杂志上发表过多篇文章。

<<系统程序员成长计划>>

书籍目录

第0章 背景知识 0.1 基础知识 0.2 开发环境第1章 从双向链表学习设计 1.1 走近专业程序员 1.2 谁动了你的隐私 1.3 Write once, run anywhere(WORA) 1.4 拥抱变化 1.5 Don't Repeat Yourself(DRY) 1.6 你的数据放在哪里第2章 写得又快又好的秘诀 2.1 好与快的关系 2.2 代码阅读法 2.3 避免常见错误 2.4 自动测试 2.5 Save your work第3章 从动态数组学习设计 3.1 动态数组与双向链表 3.2 排序 3.3 有序数组的两个应用第4章 并发与同步 4.1 并发 4.2 同步 4.3 嵌套锁与装饰模式 4.4 读写锁 4.5 无锁数据结构第5章 组合的威力 5.1 队列 5.2 栈 5.3 散列表第6章 算法与容器 6.1 容器 6.2 迭代器 6.3 动态绑定第7章 工程管理 7.1 Hello World 7.2 函数库 7.3 应用程序第8章 内存管理 8.1 共享内存 8.2 线程局部存储(TLS) 8.3 内存管理器 8.4 惯用手法 8.5 调试手段及原理第9章 从计算机的角度思考问题 9.1 变参函数的实现原理 9.2 谁在call我——backtrace的实现原理 9.3 Hello World 不能不说的十大秘密第10章 文本处理 10.1 状态机 10.2 Builder模式 10.3 管道过滤器模式第11章 分离用户界面与内部实现 11.1 分层设计 11.2 MVC架构 11.3 外壳模式第12章 撰写设计文档附录 C 语言中接口定义的不同形式

<<系统程序员成长计划>>

章节摘录

插图：假想讲给朋友听据说在代码评审时发现错误的，往往不是评审的人而是程序员自己。我也有很多这样的经历，在把自己遇到的情况讲给别人听的时候，往往是别人还没有听明白，自己已经发现里面存在的错误了。

上大学时，我常常把自己写的或者学到的东西讲给隔壁寝室的一个同学听，他说他从我这里学到了很多知识，其实我从讲的过程中，也经常会发现一些问题，对提高自己的能力大有帮助。

可惜并不是随时都能找到好的听众，幸好我们有另外一个替代办法。

记得刚开始写程序时看过一本书（忘记名字了），作者说他在写程序时，常常把思路讲给他的布娃娃听。

我没有布娃娃当听众，总不至于让我对着鼠标、键盘和显示器讲自己的思路吧，所以我会假想自己身边有个朋友，把自己的思路讲给“他”听，同时也假想“他”来质疑我。

话说回来，这种方法确实很有效，能够让思路更清晰，据说一些大师也经常使用这种方法。

这种代码阅读法会花掉你一些时间，但是可以省下更多调试时间，而且能够提高代码质量，可以说是名符其实的“写得又快又好的秘诀”之一。

至于读几遍合适，要根据情况而定，我个人觉得读两到三遍是最好的，花费的时间也不算多。

在C语言中，内存错误是最为人诟病的。

这些错误让项目延期或者被取消，引发无数的安全问题，甚至出现人命关天的灾难。

抛开这些大道理不谈，它们确实浪费了我们大量时间，这些错误引发的是随机现象，即使有一些先进工具的帮助，为了找到重现的路径，花上几天时间也不足为怪。

如果能够在编写代码的时候避免这些错误，开发效率至少提高一倍以上，而且质量也可以提高几倍。

这里列举一些常见的内存错误，供新手参考。

内存泄露大家都知道，在堆上分配的内存，如果不再使用了，就应该把它释放掉，以便后面其他地方可以重用。

在C/C++中，内存管理器不会帮你自动回收不再使用的内存。

如果你忘了释放不再使用的内存，这些内存就不能被重用了，这就造成了内存泄露。

把内存泄露列为首位，倒不是因为它会带来多么严重的后果，而是因为它是最为常见的一类错误。

一两处内存泄露通常并不至于让程序崩溃，也不会带来逻辑上的错误，而且在进程退出时，系统会自动释放所有与该进程相关的内存（共享内存除外），所以内存泄露的后果相对来说还是比较温和的。

但是，量变会导致质变，一旦内存泄露过多以至于耗尽内存，后续内存分配将会失败，程序就可能因此而崩溃。

<<系统程序员成长计划>>

编辑推荐

《系统程序员:成长计划》：结合代码详细讲解程序开发方法汇集丰富的软件开发思想CSDN专家全新一力作在学习程序开发的过程中，你是否总是为自己遇到的一些问题头疼不已，你是否还在为写不出代码而心急如焚？

作为软件开发人员，你是否时时为自己如何成为一名合格的程序员而困惑不已？

没关系，《系统程序员:成长计划》将为你排忧解难。

这是一本介绍系统程序开发方法的书。

书中结合内容详尽的代码细致讲述了不少底层程序开发基础知识，并在逐步深入的过程中介绍了一些简单实用的应用程序，最后还讲述了一些软件工程方面的内容，内容全面，语言生动，尤其适合初涉系统程序开发的人来读，有利于他们成长为更加专业的程序员。

虽然《系统程序员:成长计划》以“系统程序员”为名，但书中所蕴含的软件开发思想和方法也同样适用于其他的软件开发领域。

各种软件开发人员、相关专业的在校学生以及软件开发爱好者也都不妨读读《系统程序员:成长计划》，来分享作者多年来在学习和实践中所总结的开发方法与所领悟的开发思想。

<<系统程序员成长计划>>

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>