个人资料

silverbullettt

Qing 微博

加好友 发纸条

写留言 加关注

博客等级：14
博客积分：1019
博客访问：61,345
关注人气：102
获赠金笔：2
赠出金笔：0
荣誉徽章：

相关博文

新赚钱主线已浮出水面！
严焰茗

【高清组图】大英博物馆：10
佛教慧日艺术

心机女教师设计温柔陷阱色诱家
建胆琴心2007

1. 30涨停板预测及掘金
牛散隐士

JeanPaul
罗之衣

CBA季后赛最后名额分析&n
晚池

布拉格之冬：序篇
西狼

85%含水量的吐司
黄豆豆的一家

电子设备对于孩子
黑木耳白木耳

谁欺负了表妹？
黎双富

正文 字体大小：大 中 小

给学弟学妹们的暑期书单 (2012-06-28 00:26:54)

转 载 ▼

标签： reading it 分类： 编程思绪

应于泽瀛同志的号召，我把几个还算了解的领域内的经典书籍按我认为合理的阅读顺序在这写出，供各位暑期参考。我提到的书籍很多是多次再版的经典，要买书或者借书的同学注意一下最新的版次。

操作系统

操作系统的重要性不言而喻，大三也有这门课，但授课宗旨居然是为了考研，所以课程内容落后且过于偏理论，不太具有实践价值。由于操作系统涉及的内容太多太杂，因此我建议在学习的过程中先学习纯理论，在了解操作系统的基本原理、组成以及各部分的意义后，再结合实践进行深入地学习。

理论：

理论书籍首推Andrew Tanenbaum的《现代操作系统》，原因是塔教授写书不仅行文幽默，而且条理清晰。操作系统在发展的过程中产生了许多概念，进程、线程、锁、虚拟内存、blablabla……Tanenbaum在书中解释了这些概念产生的原因，读起来倍感亲切。

备胎：计算机界教科书达人William Stallings的《操作系统：精髓与设计原理》也不错，内容详实、值得信赖，并发与锁的问题解释得很清楚。每个主题都有一小节是结合现实中的操作系统（Windows，Linux和Solaris），但我觉得这部分内容相当鸡肋……我认为学习OS初期的理论阶段不应花太多时间，因为学习OS的重点不在理论，在于实践，所以理论阶段的学习成果是：了解操作系统的组成、基本原理、以及各部分的意义，就足以指导实践学习。

实践：

1.学习操作系统的实践无非就是：1.动手写一个操作系统；2.阅读、修改开源操作系统的源代码。但这二者都建立在对计算机系统足够熟悉的基础上，因此在此之前，要先《深入理解计算机系统》，这本CMU的经典教材大家应该都知道，我就不介绍了。

2.熟悉了计算机系统的工作原理之后，在这个阶段非常推荐自己亲手写一个操作系统。这方面的书国内最好的是于渊的《**Orange'S**：一个操作系统的实现》，除此之外还可以参考Tanenbaum的《操作系统：设计与实现》。塔神在这本书中详细介绍了Minix3的实现，于渊也是参考这本书写出了Orange'S。

3. 当然，也不是非得亲手写一个OS才能继续学习OS，我只是建议而已。到了这个阶段，直接阅读或修改开源操作系统的源代码也是极好的实践。由于开源、资料丰富、前景火爆、设计优良等诸多原因，大部分人（包括我）都会选择学习Linux，我也推荐Linux，作为一个开源操作系统，Linux的生命力非常惊人……并且有许多很好的学习资料。作为入门，首先应该看Robert Love的《**Linux**内核设计与实现》，这本书能让读者迅速进入Linux的内核世界。接下来就是德国帅哥Wolfgang的《深入**Linux**内核架构》，这本书上千页，读完之后应该对Linux的绝大部分主要功能的工作机理了然于胸了。当然，这时候书只是起指导作用，学习应以阅读源代码为主。

算法

与操作系统一样，算法的学习也分为理论和实践两部分，不同的是，算法更重于理论的分析和设计，当然编码实践也非常重要。

1. 对于初学者，首推UC Berkeley的教材《算法概论》，这是我超爱的一本算法书，它最出彩的一点是引导读者自己去发现算法。并且这本书非常薄，书中只介绍了数量非常有限的经典算法，而

更多>>

精彩图文

把重点放在算法本身，适合入门。

2. 然后是Robert Sedgewick的"Algorithms"系列，即"***Algorithms in C***"，"***Algorithms in C++***"和"***Algorithms in Java***"。Sedgewick的书最大的亮点是每一个算法都有代码实现，并且代码写得非常精彩。作为一位杰出的算法大师，Sedgewick本人对算法具有深刻的洞察力，细细阅读他的书就能体会到。学习算法是必须实践和运用的，否则即使你知道世界上最快的排序也白搭。"Algorithms"系列的每一章后面都有丰富的习题（第五部分的《图算法》才300页就有700多道习题），尽量多做习题，对锻炼编程很有好处。

3. 接下来就是Jon Bentley的《编程珠玑》。这本书绝对堪称“程序员面试宝典”，当然你要是真把它当成面试宝典来读未免太浪费。

4. 如果你读完了上述几本书，做了其中大部分习题，并且还不满足的话，那么少年，请向D. E. Knuth的"***The Art of Computer Programming***"发起挑战吧！

程序设计语言原理

工欲善其事，必先利其器。连编程语言都驾驭不好，如何当好程序员？编程语言说到底是程序员表达思想的工具，一般认为，人们思考问题的深度受到他们思考时所用语言的表达能力的限制。程序员开发软件所用的语言对他们所用的控制结构、数据结构和抽象层次也形成限制，了解更多的程序设计语言的特性能够在开发软件时减少这些限制。程序员学会新的语言结构后，能够提升开发软件时思维过程的层次。

掌握好程序设计语言的原理，不仅能帮助程序员很快的掌握新的语言，更重要的意义是更深刻地看待编程。这个主题仍然分为理论和实践两部分。

理论：

1.Robert W. Sebesta的《程序设计语言概念》比较基础，难度也不大。我大一时候读的是第6版，现在已经出到第9版了，也说明本书的经典以及编程语言发展之快速。这本书涵盖了语法和语义、名字绑定和作用域、数据类型、子程序、命令式、函数式、逻辑式等等内容，读完本书能帮助今后更系统和快速地学习新的编程语言。

2. 我现在正在读的Michael Scott的《程序设计语言——实践之路》可以说是上面那本书的全面升级版（900页，略厚），包含了上面提到的所有主题，还多了并发和脚本语言，并且还有不少关于编程语言实现的内容（书名中的“实践之路”就是这个意思）。读这本书最大的收获不是学习到更多的语言特性或者抽象模型，而是改变了看待编程语言的视角：以往是使用者，而以后能够试着从设计者的角度去看待一门语言。

实践：

了解编程语言本身最好的实践莫过于亲自动手实现编程语言，我发现不必看完大部头的《编译原理》，自己也是能写出编译器的，现在就在做这个事，待以后有了足够的经验之后，再撰文讲这方面的东西。多学习几门编程语言也是很好的实践，但是注意一点：“不能改变编程思想的编程语言，不值得学习”。至于要学习什么语言？读了上述两本书，你就心里有数了。

编程方法

这是最难通过读书来提高的能力，必须通过实践、实践和实践来提高。但是仍然有一些书能指导实践。

1. MIT的神书《计算机程序的构造和解释》（即SICP），这是我最最喜欢的专业书，无可替代，基本上我见谁都推荐……各种赞美之词详见豆瓣http://book.douban.com/subject/1148282/，我就不多说了~唯一一句就是：这本书教会了我分析和并通过编程去解决问题的能力；不好意思，还有一句：习题必做！

2. "***Concepts, Techniques, and Models of Computer Programming***"，SICP的现代升级版，有更多更新的编程模型和编程范式。引用我两年前写的评论：“<Concepts,Techniques, and Models of Computer Programming>果然是本与众不同的书.这是一本真正教你What the programming includes的书.作者在序言里提出Programming Both science and technology. "Knowing the tools prepares the student for the present. Knowing the concepts prepares the student for future developments." 在编程当中,科学和技术二者缺一不可,这样的观点还是第一次见~而且书虽然成于04年,但是书中的技术都相当新,分布式,多线程,并发,以及多范式都被当做学生应该掌握的概念写入书中,这些技术也都在最近一两年纷纷大热,作者高瞻远瞩可见一斑.而这本书的教学语言Oz更是多范式的极致,Oz受到Haskell(函数式,FP), Java(面向对象,OO), Prolog(逻辑编程,logic), Erlang(并发,concurrent)这些语言的影响,也涵盖了以上那些范式.

目前而言,各种单一编程模式都是有局限性的,掌握不同的模式并综合运用才是更好的方式.”

查看更多>>

推荐博文

安倍首相的“反省”从何而来？

【迪拜游摄】中东地区的文化名城

被催婚？我来告诉你最适合什么人

天皇道歉了吗？

美国新墨西哥州空中惊现“冰光环

探秘西太平洋“幽灵墓地”（组图

2015年最火的科技公司

【童星】十大欧美电影中的惊艳童

马年那个生机勃发的仲夏——东北

带着13个月宝宝去三亚（内附详


寻找撒尿小孩儿于连


美女探寻外星人38年


鲨鱼爱攻击男性冲浪者


非洲雄狮撕咬河马尸体


揭秘世界最毒死亡之蛙


40岁的灵魂歌者顺子

查看更多>>

这本书目前还没有中文版，不过很好读的~

计算理论

学习计算理论基本上对程序员没有直接的帮助，但是从兴趣以及长远发展的角度来说，我推荐。并且（一）这些是计算机科学的根本，没有它们计算机科学不能算作是个正经学问，因此，一个自称计算机科学专业的人，应当知道这些；（二）这些是美好的，值得在短暂的人生中去经历去见识（etone语）。

1.《逻辑的引擎》，这不是教科书，而是一本介绍逻辑学历史以及对数理逻辑做出重大贡献的数学家的书。作者Martin Davids也是一位杰出的数学家，还是Alonzo Church的博士生，由他来写这本书也再适合不过了。介绍这本书，是希望能通过看Leibniz、Frege、David Hilbert、Alan Turing、Alonzo Church、Von Neumann……这一颗颗闪耀的明星如何改变这个学科，点燃大家对数理逻辑的热情。Martin Davids的文笔很赞！

2.我最喜欢的计算理论入门书是Michael Sipser教授的《计算理论导引》，我没有读最后3章，但在前面章节的阅读过程中，感觉接受了一次严谨证明、神奇思想与美妙方法的洗礼。这本书最美妙之处，并不在于它描述的那些艰深的问题和结果，而是它全面展示了计算理论中 那些最引人入胜的深刻想法：抽象问题的形式语言描述，确定性和非确定性的引入，对角化和停机问题，归约的思想，完全性（etone语）……每一个定理都从思路出发，引导读者领略计算机科学的神奇。

3.机缘巧合，某日在网上找到一本"***Foundations of Computer Science, C Edition***"，作者是大名鼎鼎的Alfred V. Aho和Jeffrey D. Ullman，也就是撰写“龙书”的那二位大哥，他们把这本书在互联网上公开（Ullman教授的主页可以找到）。这本书更适合工科学生，因为它没有那么多枯燥的理论，而是从实践出发去引出背后的理论模型（否则书名也不会有个"C Edition"了），最枯燥的谓词逻辑放在全书的最后。所以既想学习计算理论又怕枯燥的同学可以看看这本书。还有，它也没有中文版……

本来只是推荐些书，没想到写了不少废话，如果你不喜欢听我啰嗦，只看粗体字就可以。这有好几本书是一个暑假都看不完的，像其中最厚的几本我也是到现在也没读完，所以希望大家能坚持。

祝学习愉快^-^

程亦超

mark

2012-6-28 21:33 回复(0)

dawnstar

博主好厉害，然后发现跟博主是同龄人呢！ 关注你微博了，这个是我http://weibo.com/2096528123

2012-7-28 17:37 回复(1)

罗杰亚克斯曼

marked

2014-7-16 13:00 回复(0)

罗杰亚克斯曼

marked

2014-7-16 13:02 回复(0)

发评论

git65161：您还未开通博客，点击一秒开通。

分享到微博 评论并转载此博文 匿名评论

验证码： 请点击后输入验证码 收听验证码

发评论

前一篇 天才程序员

后一篇 奇妙的层次