HZ BOOKS
华章科技

一本影响了全球数百万程序员的求职面试宝典，Amazon超级畅销书，持续销售近10年

授人以鱼，全面讲解程序员面试时需要掌握的各种编程必备知识和技巧；授人以渔，针对来自全球顶尖IT企业的极具代表性的面试题给出了解答思路，并揭示了这些企业的面试过程

WILEY

程序员面试攻略

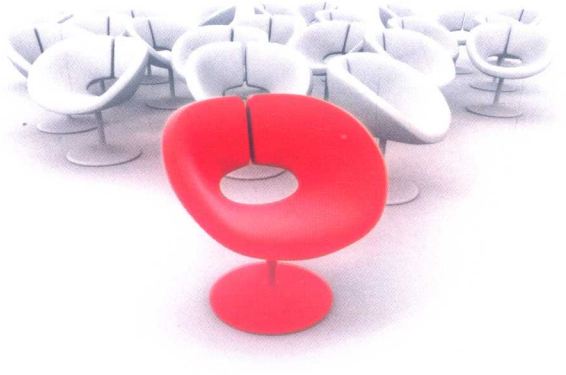
（原书第3版）

（美）John Mongan Eric Giguère Noah Kindler 著 李秉义 译

Programming Interviews Exposed

Secrets to Landing Your Next Job

Third Edition



机械工业出版社
China Machine Press

3

程序员面试攻略

（原书第3版）

（美）John Mongan Eric Giguère Noah Kindler 著 李秉义 译

Programming Interviews Exposed

Secrets to Landing Your Next Job

Third Edition

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

程序员面试攻略（原书第3版）/（美）蒙干（Mongan, J.）等著；李秉义译. —北京：机械工业出版社，2013.11

书名原文：Programming Interviews Exposed: Secrets to Landing Your Next Job, Third Edition

ISBN 978-7-111-44434-3

Ⅰ. 程… Ⅱ. ①蒙… ②李… Ⅲ. 程序设计-工程技术人员-资格考试-自学参考资料 Ⅳ. TP311.1

中国版本图书馆CIP数据核字（2013）第245158号

本书版权登记号：图字：01-2013-0222

这是一本影响了全球数百万程序员的求职面试宝典，Amazon超级畅销书，持续销售近10年，好评如潮。它授人以鱼，全面讲解程序员面试时需要掌握的各种编程必备知识和技巧；同时也授人以渔，对来自全球顶尖IT企业的极具代表性的面试题给出了解答思路，并揭示了这些企业的面试过程，帮助求职者在面试中应付自如。

本书深入阐释了程序员在求职过程所面临的求职沟通和程序设计技能两个方面的问题。在程序设计技能方面，本书介绍了作为一名程序员必须具备的一些基本功，包括链表、树和图、数组和字符串、递归、排序、并发、面向对象编程、设计模式、数据库、图形学与位操作等。强调沟通方法是本书的亮点，本书包括如何编写一份重点突出的个人简历，如何与猎头公司和人力资源代表打交道，在面试时如何与面试官进行沟通等内容。本书中的面试题除了有详细解析和答案外，还对相关知识点进行了扩展说明。通过对丰富的面试题目应用循序渐进的解答方法来模拟面试过程，以强化学到的技能，真正做到由点成线，举一反三，对读者从求职就业到提升计算机专业知识都有显著帮助。

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：秦 健

北京市荣盛彩色印刷有限公司印刷

2014年1月第1版第1次印刷

186mm×240mm · 17印张

标准书号：ISBN 978-7-111-44434-3

定 价：59.00元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：（010）88378991 88361066 投稿热线：（010）88379604

购书热线：（010）68326294 88379649 68995259 读者信箱：hzjsj@hzbook.com

译 者 序

顶级软件公司的程序设计面试通常是很艰难的，即使是非常优秀的程序员，如果没有经历过这类面试并进行充分的准备，也很难通过。拥有扎实的编程技能和丰富的计算机基础知识通常还不足以让你轻松通过这类面试，你还需要掌握面试中的沟通技巧与答题技巧。这样才能在有限的面试时间中让面试官了解你的真实能力。

本书深入探究了顶级软件公司的程序设计面试过程，并且对可能出现的面试题目进行了分类汇总，引导你学习解决各类问题的方法。而不是通过罗列大量面试问题，通过死记硬背来教你准备程序设计面试。此外，由于程序设计题目千变万化，通过死记硬背不能给你通过面试带来实质性的帮助。

努力学完本书，你将能够掌握顶级软件公司的面试过程，在面对各类面试问题时能够迅速找到切入点，从容应对，并在面试过程中将自己的能力充分展示给面试官。

感谢机械工业出版社的编辑在本书的翻译工作中所付出的努力。尽管译者在翻译时反复推敲，查阅了大量资料以确保专业术语的翻译准确，但译文中难免还有不当之处，敬请批评指正。

前　言

我们首先要告诉你的最重要的事情与第 1 版相同：你对本书钻研得越深，收获就越大。如果你只是泛泛地阅读本书，你会学到一些东西，但是如果在看答案前先自己尝试解决问题，那么你会收获更多。

本书第 1 版出版已经超过十年了，这就是说，许多我们必须告诉你的事情已经改变，因此需要有个机会来改编本书。

第 3 版进行了最大程度的更新。除了对前一版进行改编、扩展以及更新，还增加了关于排序和设计模式这些重要的主题。本书对非编程部分的内容也进行了改编，以反映如今就业市场的现实。所有这些我们都像前一版一样，保持容易理解的风格和循序渐进的思考过程。

样例代码大部分采用 C、C++ 或 Java，但在大多数情况下数据结构和算法才是重点，使用什么语言实现是次要的。所有的样例对于一个有经验的程序员来说应该都是很容易理解的。

我们中的一员（Eric）最近通过了他梦想中的 Google 公司职位的面试，这增加了他对于程序设计面试的认识，并将其收获反映在了这一版中。

我们希望你喜欢第 3 版，并且希望对你获得自己喜欢的职位有所帮助。我们非常希望听到你关于本书的想法以及你的面试经验。你可以通过 authors@piexposed.com 与我们联系。访问本书的官方网站：http://www.piexposed.com 可以获得最新的更新以及更多信息。

第 1 版前言

如果你像我们一样，通常就不会读前言。但这本书的前言很有用，因此希望你能破例一次好好读一下。如果你对前言确实不感兴趣，我们希望你能看完下面这句话：你对本书钻研得越深，收获就越大。如果你只是泛泛地阅读本书，你会学到一些东西，但是如果在看答案之前先自己尝试解决问题，那么你会收获更多。

当你应聘一份程序设计、软件开发或技术咨询方面的工作时，雇主几乎总是会安排一次面试来考察你的程序设计能力。我们写作本书的目的就是希望能够帮助大家顺利地通过这类面试。程序设计面试的目的是为了考察应聘者的程序设计水平和动手能力，其过程与传统意义上的求职面试并没

有多少相似之处，所以传统的应聘秘笈和面试技巧在这里并没有多大用处。程序设计面试题以程序设计问题、智力难题以及各种与计算机有关的技术为主。本书将对这些常见的程序设计面试题进行讨论，并通过一些取材真实的面试题向大家展示如何才能最好地回答这些问题。

看到这里，你可能会产生这样的疑问：作为本书的作者，我们都是些什么人？是谁让我们写这本书的？我们两人都是刚毕业不久的研究生，在过去的几年里，我们参加了许多场面试。从老牌大公司的技术咨询职位到新兴公司的编写设备驱动程序的职位，我们面试过的工作岗位可以说五花八门，这本书就是我们根据自己亲身参加过的各种面试（有成功也有失败）总结出来的。我们认为，这是写作本书的基础。说实话，我们并不清楚负责招聘工作的人力资源部门主管对程序设计面试工作都有哪些想法，我们也不清楚猎头公司将依据什么样的标准来评估应聘人员在程序设计面试中的表现。但在读完本书之后，相信大家都会对美国顶尖的软件和计算机公司里的程序设计面试情况有一个比较实际的了解，并知道自己应该去做些什么来赢得自己想要的那份工作。

注意

根据自己的亲身经历，我们认为现在的程序设计面试工作还有许多地方需要改进。现在的做法过于偏重考察应聘人员解答智力难题或者类似问题的能力，而忽视了对其知识面和知识深度的考察，因此很多在软件行业中取得成功所必须具备的重要素质都无法得到准确衡量和评估。

需要特别说明的是，本书中的面试题目没有一道是我们虚构出来的，这些题目全部来自于我们此前参加过的面试。换句话说，类似的题目类型和难度很可能会出现在你今后参加的某次程序设计面试中。但大家同时也要明白，这本书里的问题只是程序设计面试中一些比较有代表性的题目，而不是一个包罗万象的习题集。如果希望靠死记硬背本书的例题和答案来通过程序设计面试，就很可能弄巧成拙。在面试中你可能会遇到这本书里的问题，但你不能抱有这样的幻想。程序设计面试题本来就是千变万化的，而一位聪明的面试官在看过本书之后，肯定不会再用本书中的题目。可话又说回来，程序设计面试题范围内的类型也就那么多，想变化也变化不到哪里去。只要你不是死抠本书中给出的例题，而是对它们所代表的试题类型进行研究，那么无论在程序设计面试中遇到什么样的题目，你肯定都能应付自如。

为了帮助大家提高解决问题的能力，我们采用了一种循序渐进的办法。首先，根据实际情况，我们会在给出面试题之前对有关的重要概念进行复习。其次，我们会把解决问题的整个思路向大家解释清楚，而不是简单地直接给出问题的答案。我们发现，从例题的使用方面来讲，本书以外的其他教科书或习题集几乎全都采用了另一种套路：先列出一个问题，接着马上给出答案，然后再解释那个答案为什么是正确的。以我们个人的体会来说，这种套路往往不能给你以最大的帮助：你能够看懂某个具体的答案，并知道它为什么是正确的，但很难了解和把握作者得出有关答案的思路，不

容易在遇到与例题类似的问题时做出正确的分析和判断。为了避免上述弊病，本书采用了一种循序渐进的解题方法，而我们希望本书中的解题思路不仅能够让大家知道什么是正确的解决方案，还能让大家明白怎样才能得出正确的解决方案。

只看不动手是学不到真本领的。如果你想从本书中得到的收获最大，就必须亲自动手去尝试解决书中的每一道题。我们建议大家采用下面的学习方法：

（1）看过题目之后，先把书放在一边，自己动手去寻找答案。

（2）如果你中途卡住了，再回过头来研究书中的解决方案。为了让大家开动脑筋，这本书里所有例题的答案都安排在有关内容的最末尾，所以你完全不必担心我们会提前“泄密”，让大家“意外地”看到答案。

（3）在看过足够多的内容并得到足够多的提示之后，你应该再次把本书放在一边，继续开动脑筋。

（4）如此重复，直到彻底解决某个问题为止。

你通过自己的努力而得出的解决方案越多，对有关问题的理解也就越透彻。这种学习方法还有另外一个好处，那就是它与程序设计面试的实际过程非常相似：你必须完全依靠自己来解决面试问题，但在需要的时候，面试官会给出必要的提示。

程序设计是一种难度极大的技术性艺术，只通过一本书就把计算机科学和程序设计工作所涉及的各种细节全都介绍给大家是根本不可能的。因此阅读本书需要有一定基础。我们希望你的计算机知识不低于大学计算机系一年级或二年级学生的水平。具体地说，我们希望你能够熟练地使用 C 语言进行编程，有使用 C++ 或 Java 进行面向对象编程的经验，了解计算机体系结构和计算机科学理论方面的基础内容。这是参加程序开发工作最小的要求，所以大多数面试官都会有类似的预期。如果发现自己在上述几个方面有所欠缺，请务必在找工作和参加程序设计面试之前把功课补上。

在本书的读者中，肯定会有很多人在计算机方面的学识与经验大大超出我们刚才提出的最小要求。如果你就是其中一员，那么你可能对本书中的一些高级主题，比如数据库、图形处理、并发计算以及 Perl 语言等更感兴趣。但千万不要因为自己的经验比较丰富就忽视了基础性的概念和试题。不管你的简历写得多好，面试官仍会从最基本的问题开始提问。

我们已经尽了最大的努力来保证这本书里的信息是正确无误的。所有的程序代码都经过了编译和测试。但就与读者在你们自己的程序设计工作中遇到的情况一样，程序设计漏洞和错误是在所难免的。一旦发现或者得知此类错误，我们会立刻把它们公布在 http://www.piexposed.com 上。

我们相信，在你寻找新工作时，本书对你一定有所帮助。同时，我们还希望本书中的面试题目分析和解决方案能够对你找工作有所启发。如果你想把你的读后感、对每个问题的具体看法，或最近看到的某个程序设计面试问题告诉我们，我们将非常欢迎。我们的电子邮件是：authors@piexposed.com。

预祝大家都能找到一份满意的工作！

致　谢

第 3 版的出版非常不寻常，我们感谢来自 Wiley 出版社的工作人员所付出的努力，使得这个版本可以及时成功地顺利完成。可以快速克服任何新障碍的编辑 Maureen Spears，个人对此关注的出版人 Jim Minatel，以及特别关键的高级组稿编辑 Carol Long，我们感谢他们为此付出的时间、工作以及帮助。

这个版本的质量因为 Wayne Heym 的建议以及详细的审阅而得到巨大提升，我们感谢他提供的大量帮助。

此外，John 非常感谢 Michael J. Mongan 的帮助，协助他可以参与这个版本的工作。

如果没有前两版，以及许许多多为前两版作出贡献的人们，是不可能有第 3 版的。因此，我们也感谢原来的编辑 Margaret Hendrey 和 Marjorie Spencer，感谢他们的耐心和帮助。同时也感谢原来的审稿人和顾问 Dan Hill、Elise Lipkowitz、Charity Lu、Rob Maguire 和 Tom Mongan。Dan 的贡献最多，第 1 版的质量由于他认真细致的审阅而得到了巨大的提升。

技术编辑简介

Michael Gilbert 长期担任几家不同的工程公司的系统程序员。在 Atari ST 公司开始开发游戏，并且是《 Start 》杂志的特约编辑。多年来，他为来自全球范围的客户开发 PC 上和 Mac 上的游戏软件。同时他也是一位专家级的 Flash Actionscript 程序员，他研发了一个流行的网络游戏平台 HigherGames，你可以在 www.highergames.com 上看到。现在他喜欢上了开发 iPhone 和 iPad 上的游戏，目前在 AppStore 中有 4 款游戏（ Wóridgo、Jumpin' Java、Kings Battlefield 以及 Set Pro HD）。在业余时间，他喜欢在一个友好的拼字游戏中与他的妻子 Janeen 战斗。你可以在 Twitter 上收听他：@mjia711。

Justin Vogt 是一位经验丰富的职业软件开发者，具有很多综合能力（技术、架构、设计、沟通、创新、管理以及领导研发）。他具有超过 15 年的丰富软件研发经验，曾经参与过的项目包括嵌入式软件、移动开发、网络开发、商业软件开发、通信设备、医疗应用程序和非营利组织解决方案的开发。

目　录

第0章
引　言

获得一份非常好的编程工作不能靠运气，而是要靠准备工作。如今大多数软件公司的程序设计面试过程都经过仔细设计，用来确定你是否真的可以编程。这会是一个艰难的过程，尤其是面试形式多种多样，使得这个过程完全不同于你在学校或其他工作中所经历过的面试。如果你之前从未经历过程序设计面试，那么这将会是个沉重的打击。即使是非常优秀的程序员，如果没有经历过程序设计面试，也没有为面试做好准备，也会相当艰难。

写这本书的目的是帮助你准备这个技术面试过程，使得你在展示自己是个多么优秀的程序员时没有任何问题。本书并不教你如何编程，而是展示在程序设计面试中如何通过你的编程技能快速脱颖而出。在你阅读本书时，请记住，大部分程序设计面试事实上并不是记忆力测试，所以本书并不提供在面试中需要记住的小抄，而是通过实例教你通过面试所需的技巧和思考过程。掌握这些最好的办法是花时间解决这些问题，并理解它们。如果你这样做了，在面对面试问题时，你将会非常自信，因为你已经为解决任何面试问题做好了准备，这使得你更加接近你希望的职位。

为什么要进行程序设计面试

为什么软件公司都会进行程序设计面试？他们希望雇佣那些优秀的、可以与他人合作并能成功创造出优秀产品的程序员。但遗憾的是，经验表明，相当一部分应聘程序员职位的应聘者都不会编程。你也许认为通过认真地检阅应聘者的简历、经验、所学课程以及学位可以筛选出来，但事实上这通常是无效的。有相当数量的应聘者拥有出众的简历、业界多年的相关经验，但是连一些简单的编程任务都无法胜任。他们中的许多人都收集了足够多的术语，在讨论编程技术的话题时看上去非常有竞争力。雇佣这些不会编程的开发人员很容易击垮一个部门（甚至是一个小公司）。

由于认识到普通的面试过程对于区分应聘者是否会编程是无效的，所以雇主们通常采取

以下步骤：在面试中让应聘者编写一些程序。程序设计面试就诞生了。程序设计面试在区分应聘者是否会编程时非常有效，这也正是技术面试过程中几乎都有这一部分的原因。

程序设计面试中困难的部分是雇主不仅想知道哪些人不会编程，而且想从应聘者中挑选出优秀的程序员，这更难。通常，面试官通过使用困难的、有挑战的编程题目，并留意应聘者解决它们的速度和准确性来衡量应聘者的能力。

这种方法的问题在于面试的时间有限，在程序设计面试过程中仅可以测试部分技能，这些技能只有部分与现实世界的开发有关。在程序设计面试中必须让你当场解决问题，有个人看着你，并且没有任何可以参考的东西。没有时间来编写大量代码，所以问题必须有简短的解法。大多数有简短解法的问题都很简单，为了避免这种情况，许多面试问题都涉及不常见的算法技巧、随意的限制或者奇怪的语言特性。因为这类问题在现实世界的开发中并不常见，即使一个优秀的程序员如果没有经历过这种奇怪的面试通常也是无法通过的。

相反，许多在专业开发环境中的核心技能在程序设计面试中却没有得到很好的评估，甚至未进行评估。这些技能包括沟通、合作、架构、对大型代码库的管理、时间管理、按时间表持续地产出可靠的代码，以及处理大型项目、区分各部分的组件，并且推动整个项目的完成等能力。

显然，程序设计面试并没有提供衡量一个应聘者作为公司未来员工的价值的完美方法。但是就像丘吉尔对于民主的衡量方法所说的一样，程序设计面试是技术面试中最差的方法，除了其他已经被尝试过的方法之外。更重要的一点是程序设计面试是雇主选择到底雇佣谁的方法，所以无论这种衡量方法是否完美，你都需要在其中表现良好。本书就是为了展示如何将你的编程技能应用于各种奇怪的面试问题，并且帮你准备和进行练习，以在你想要的职位面试中脱颖而出。

如何使用本书

进行准备是掌控面试进程的关键。下面是如何使用本书来准备程序设计面试的一些一般性方法。

- **留出充足的时间进行准备**。尽可能早地开始准备，最好是在参加面试几周前或一个月前。你需要时间来练习这里介绍的概念。(如果你没有这么多时间，试着找一些不被打扰的大块时间来学习这些内容。)
- **练习回答问题**。不要只是通读问题的解法。在解决问题卡住时，将问题的解法作为提示，并且用来验证你的答案。试着模拟面试过程。大多数情况下，你需要在一个白板上或纸上写出代码，这需要练习！这听上去很傻，但是确实需要进行一些练习，以便在编程的时候你可以用一支笔来代替键盘。
- **确保你理解问题背后的概念**。理解问题背后的概念是成功的关键。不要跳过或掩盖你不理解的东西。这本书提供了足够的解释让你复习之前学习过的主题，但是如果

你遇到一些已经完全忘记的东西或从未学习过的东西时，你可能需要读一些其他的参考资料。

- **不要浪费时间死记硬背问题的答案**。面试官不太可能用本书中的任何问题来面试。即使他们这样做，也会对这些问题进行一些修改。如果你仅仅把问题的答案背下来，那么你的答案很可能是不正确的。
- **继续练习**。不要在完成本书中的面试题目后停止准备。继续练习一些编程问题。从网络上很容易找到一些问题。再寻找一些参考资料，特别是你擅长的领域，并继续阅读。

你的健康和福祉是你最重要的资本，它会影响你的学习和面试。记住需要足够的睡眠(特别是接近面试日期的时候)，并且要保持合理的锻炼和饮食。定时进行休息有助于你的头脑整合所有的资料，不要在临近面试的最后一刻还不停地学习。如果放松一下，使自己头脑清晰，这要比在考前不停地学习更高效。

作为准备的一部分，请访问网站 http://www.piexposed.com，注册并获取我们的邮件列表，了解我们为帮助你准备面试开发的智能手机应用。

现在让我们开始吧！