

Beauty of Programming

Broadview®
www.broadview.com.cn



编程之美

— 微软技术面试心得

《编程之美》小组 著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

1101

编程之美

——微软技术面试心得

《编程之美》小组 著



电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书收集了约 60 道算法和程序设计的题目，这些题目大部分在微软的笔试、面试中出现过，有的曾被微软员工热烈地讨论过。作者试图从书中各种有趣的问题出发，引导读者发现问题、分析问题、解决问题，寻找更优的解法。本书内容分为以下几个部分。

游戏之乐：从游戏和其他有趣问题出发，化繁为简，分析总结。

数字之魅：编程的过程实际上就是和数字及字符打交道的过程。这一部分收集了一些好玩的对数字进行处理的题目。

结构之法：汇集了常见的对字符串、链表、队列以及树等进行操作的题目。

数学之趣：列举了一些不需要写具体程序的数学问题，锻炼读者的抽象思维能力。

书中绝大部分题目都提供了详细的讲解。每道题目后面还有一至两道扩展问题，供读者进一步钻研。

书中还讲述了面试的各种小故事，告诉读者微软需要什么样的技术人才，重视什么样的能力，如何甄别人才，并回答了读者关于 IT 业面试、招聘、职业发展的疑问。这本书的很多题目会出现在 IT 行业的各种笔试和面试中，但本书更深层的意义在于引导读者思考，和读者共享思考之乐、编程之美。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目 (CIP) 数据

编程之美：微软技术面试心得 / 《编程之美》小组著. —北京：电子工业出版社，2018.11

ISBN 978-7-121-33782-6

I. ①编… II. ①编… III. ①程序设计 IV. ①TP311.1

中国版本图书馆 CIP 数据核字(2018)第 038905 号

责任编辑：刘 皎

印 刷：三河市双峰印刷装订有限公司

装 订：三河市双峰印刷装订有限公司

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×980 1/16 印张：22.75 字数：400 千字

版 次：2018 年 11 月第 1 版

印 次：2018 年 11 月第 2 次印刷

定 价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888，88258888。

质量投诉请发邮件至 zltts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

推荐序

我在卡内基梅隆大学毕业找工作的时候，经常和其他同学一起交流面试的经验。当时令求职者“闻面色变”的公司有微软，研究所有 DEC 的 SRC。每次有同学去微软或 SRC 面试，回来的时候都会被其他同学追问有没有什么有趣的面试题。我也是那时第一次听说“下水道井盖为什么是圆的”这一问题。

我自己申请加入微软美国研究院时被面试了两天，见了 15 个人，感觉压力很大。至今还记得当有一位面试者不断追问我论文中一个算法的收敛性时，我们进行了热烈讨论。在微软工作的十几年中，我自己也面试了非常多的新员工。特别在微软亚洲研究院的 9 年，经常感觉很多刚刚毕业的优秀学生基础很好，但面试的准备不足。我非常欣慰地看到邹欣工程师和微软亚洲研究院其他同事们努力编写了这本好书，和大家一起分享微软的面试心得和编程技巧。相信更多的同学会因此成为“笔霸”、“面霸”，甚至“offer 霸”。

程序虽然很难写，却很美妙。要想把程序写好，需要学好一定的基础知识，包括编程语言、数据结构和算法。程序写得好的通常都有缜密的逻辑思维能力和良好的数理基础，而且熟悉编程环境和编程工具。古人说“见文如见人”，我觉得程序同样也能反映出一个人的功力和风格，好的程序读来非常赏心悦目。我以前常出一道面试题是“展示一段自己觉得写过的最好的程序”。

编程很艰苦，但是很有趣。本书的作者们从游戏中遇到的编程问题谈起，介绍了数字和字符串中的很多技巧，探索了数据结构的窍门，还发掘了数学游戏的乐趣。我希望读者

在阅读本书时能找到编程的快乐，欣赏到编程之美。本书适合计算机学院、软件学院、信息学院高年级本科生、研究生作为软件开发的参考教材，也是程序员继续进修的优秀阅读材料，更是每位申请微软公司和其他公司软件工程师之职的面试必读秘笈。

人类的生活因为优秀的程序员和美妙的程序而变得更加美好。

沈向洋

微软公司杰出工程师

微软公司全球资深副总裁

2008年春节于香港

序

一位应聘者 (interviewee) 在我面前写下了这样的几行程序：

```
while (true) {  
    if (busy) i++;  
    else  
}
```

然后就陷入了沉思，良久，她问道：“那 else 怎么办？怎么能让电脑不做事情呢？”

我说：“对呀，怎么才能让电脑闲下来？你平时上课、玩电脑的时候有没有想过？这样吧，你可以上网查查资料。”

她很快地在搜索引擎中输入“50% CPU 占用率”等关键字，但是搜索并没有返回什么有用的结果。

在她忙着搜索的时候，我又看了一遍她的简历，从简历上可以看到她的成绩不错，她学习了很多程序设计语言，也研究过“设计模式”“架构”“SOA”等，她对 Windows、Linux 也很熟悉。我的面试问题是：“如何写一个短小的程序，让 Windows 的任务管理器显示 CPU 的占用率为 50%？”这位应聘者尝试了一些方法，但是始终没有写出一个完整的程序。面试的时间到了，她看起来比较遗憾，我也一样，因为我还有一系列的后续问题没有机会问她：

- 如何能通过命令行参数，让 CPU 的使用率保持在任意位置，如 90%？
- 如何能让 CPU 的使用率表现为一条正弦曲线？
- 如果你的电脑是双核(dual-core CPU)的，那么你的程序会有什么样的结果？为什么？

自从 2005 年回到微软亚洲研究院后，我面试过不少应聘者，作为面试者，我最希望看到应聘者给出独具匠心的回答，这样我也能从中学到一些“妙招”。遗憾的是看到“妙招”的时候并不多。

我也为微软校园招聘出过考题，走访过不少软件学院，还为员工和实习生做过培训。我了解到不少同学认为软件开发的工作没意思，是“IT 民工”“软件蓝领”。我和其他同事也听到一些抱怨，说一些高校计算机科学的教育只停留在原理上，忽视了对原理和技术理解和运用。

写程序真的没有意思吗？为什么许多微软的员工和软件业界的牛人乐此不疲？我和一些喜欢编程的同事和实习生创作编写了这本书，我们希望通过分析微软面试中经常出现的题目，来展示编程的乐趣。编程的乐趣在于探索，而不是在于背答案。面试的过程就是展现分析能力、探索能力的过程，在面试中展现出来的巧妙的思路、简明的算法、严谨的数学分析就是我们这本书要谈的“编程之美”。

有时候会有同学问：“你们是不是有面试题库？”言下之意是每个应聘者都是从“库”中随机抽出一道题目，如果答对了，就中了；如果答错了，就 bye-bye 了。书中有一些关于面试的问答，我想它们可以回答这样一些疑惑。

本书的题目，一部分源自各位作者平时想出来的，例如，有一次一位应聘者滔滔不绝地讲述自己如何在某大型项目进行 CPU 的压力测试，听上去水分不少，我一边听一边琢磨“怎样才能考察一个人是否真正懂了 CPU，任务调度……”，后来就有了上面提到的“CPU 使用率”的面试题。有些题目来自于平时的实践和讨论，比如一些和游戏相关的题目。有些题目是随手拈来，比如我看到朋友的博客上有一道面试题，自己做了一下，发现自己的解法并不是最优的，但是，倒是可以作为一个面试题的题目，第 2 章的“程序理解和时间分析”就是这么得来的。书中有些题目在网上流传较广，但是网上流传的解法并不是正解，我们在书中加上了详细的分析，并提出了一些扩展问题。还有一些题目在教科书和专业书籍中有更深入的分析 and 解答，读者可以参考。

书中的大多数题目都能在 45 分钟内解决，这也是微软一次技术面试的时间。本书不是一个“答案汇编”，很多题目并没有给出完整的答案，有些题目还有更多的问题要读者去解答，这是本书和其他书籍不一样的地方。面试不是闭卷考试，如果大家都背好了“井盖为什么是圆的”的答案来面试，但是却不会变通，那结果肯定是令人失望的。

为了方便读者评估自己的水平，我们还按照每道题目的难度制定了相应的“星级”：

- 一颗星：不用查阅资料，在 20 分钟内完成；
- 两颗星：可以在 40 分钟内完成；
- 三颗星：需要查阅一些资料，在 60 分钟内完成。

由于每个人的专业背景、经历、兴趣不一样，这种“星级”仅仅是一种参考。

作者们水平有限，书中的题目并不能代表程序设计各个方面的最新进展，虽然经过几轮审核，不少解法仍可能有漏洞或错误，希望广大读者能给我们指正。我们已在微软亚洲研究院的门户网站（www.msra.cn）上开辟专栏（www.msra.cn/bop）和读者交流——初学者和高手都非常欢迎！

本书的内容分为下面几个部分。

- **游戏之乐**：电脑上的游戏是给人玩的，CPU 也可以让人“玩”。这一部分的题目从游戏和作者平时遇到的有趣问题出发，展现一些并不为人重视的问题，并且加以分析和总结。希望其中化繁为简的思路能够对读者解决其他复杂问题有所帮助。
- **数字之魅**：编程的过程实际上就是和数字及字符打交道的过程。如何提高掌控这些数字和字符的能力对提高编程能力至关重要。这一部分收集了一些好玩的对数字进行处理的题目。
- **结构之法**：对字符及常用数据结构的处理几乎是每个程序必然会涉及的问题，这一部分汇集了对常用的字符串、链表、队列以及树等进行操作的题目。
- **数学之趣**：书中还列了一些不需要写具体程序的数学问题，但是其中显示的原理和解决问题的思路对于提高思维能力还是很重要的，我们把它们单独列出来。
- **关于笔试、面试、职业选择的一些问答**：微软的面经，各种技术职位的介绍是很多学生所关心的内容，因此我们把一些相关的介绍和讨论也收录了进来。

我们希望《编程之美》的读者是：

1. 大学计算机系、软件学院或相关专业的大学生、研究生，可以把这本书当作一个习题集；
2. 面临求职笔试、面试的 IT 从业人员，不妨把这本书当作“面试真题”，演练一下；
3. 编程爱好者，平时可以随便翻翻，重温数学和编程技能，开拓思路，享受思考的乐趣。

《编程之美》由下面几位作者协同完成，如果把这本书的写作比作一个软件项目，它有下面的各个阶段，每个阶段则有不同的目标和角色。

1. 构想阶段：邹欣。
2. 计划阶段：邹欣、刘铁锋、莫瑜。
3. 实现阶段/里程碑（一）：上述全部人员，加上李东、张晓、陈远、高霖（负责封面设计）。
4. 实现阶段/里程碑（二）：上述全部人员，加上梁举、胡睿。
5. 稳定阶段：上述全部人员，加上博文视点的编辑们。
6. 发布阶段：邹欣、刘铁锋和博文视点的编辑们。

这本书从 2007 年 2 月开始构思，到 2007 年 11 月底交出完整的第一稿，花费的时间比每一位作者预想的要长得多，一方面是大家都有日常的工作和学习任务要完成；更重要的是，美的创造和提炼，是一个漫长和痛苦的过程。要把“编程之美”表达出来，不是一件容易的事，需要创造力、想象力和持久的艰苦劳作。就像沈向洋博士经常讲的一句话——Nothing replaces hard work。

这本书的各位作者，都是利用自己的业余时间参与这个项目的，他们的创造力、热情、执着和专业精神让这本书从一个模糊的构想变成了现实。通过这次合作，我从他们那里学到了很多，借此机会，我对所有参与这个项目的同仁们说一声：谢谢！

在本书编写过程中，作者们得到了微软亚洲研究院的许多同事的帮助，具体请参见“致谢”。

我们希望书中展现的题目和分析，能像海滩上美丽的石子和漂亮的贝壳那样，反映出造化之美，编程之美。

邹欣

2007 年 11 月于北京

补记

一晃《编程之美》已经出版 10 年了，在 10 年之后的今天，仍然有机会再次更新序言，唯有感谢和荣幸！

重读自己 10 年前写的文章 (<http://www.broadview.com.cn/33782>)，“简化、持续挑战、开拓”，这三个观点依然在持续践行……在过去的 10 年中，我已离开微软亚洲研究院、创办海豚浏览器、并购退出，并即将开始第二次创业旅程。每一次重大决定，都代表着我的不断挑战和开拓。

回想起当年“痛不欲生”地在邹老师的“摧残”之下，一次次迭代改进，似乎暗无天日、永远没有尽头……这段经历，已经成为我用来实证“坚持努力，做足过程，成果自现”的案例。

感谢读者的认可，感谢出版社和编辑的不懈努力，才有机会让这本书再次出发！

祝每一位读者都能挑战自己的极限，开拓一片天地！

刘铱锋

2018 年 8 月

致 谢

《编程之美》这本书从构思、编写到最后的出版，得到了许多同事和朋友的帮助。在此作者们要特别感谢以下人士。

微软亚洲研究院的多位同事热情地与我们分享了他们觉得有意思的题目，他们分别是：邓科峰、宋京民、宋江云、刘晓辉、赵爽、李劲宇、李愈胜和 Matt Scott。

感谢微软亚洲研究院技术创新组的同事梁潇、殷秋丰，他们认真审阅了所有的题目和解答，找出了不少 bug¹。技术创新组的另外几位优秀工程师李愈胜、魏颢、赵婧还帮助我们解决了书中的几个难题。

感谢研究院的同事、著名技术作家潘爱民对我们的鼓励，他审阅了全部稿件，并且提出了不少意见。

本书的封面和插图都出自研究院的实习生高霖之手，他在 10 余个构图都被否定的情况下，坚持不懈，最后拿出了“九连环”的封面设计，得到作者和出版方的一致认同。

在本书写作的过程中，作者们各自的“老板”——杨晓松、姚麒、田江森和刘激扬都给予了不少支持，在此特表示感谢。

作者们的“老板的老板”，研究院前任院长，微软公司资深副总裁沈向洋博士，现任院长洪小文博士对本书一直很关心和支持。沈向洋博士在百忙之中还亲自为本书写了序。

¹ 本书残留的 bug 都是作者们的责任。

微软亚洲研究院市场部的金俊女士、葛瑜女士对本书的推广提供了很大帮助。负责 www.msra.cn 网站的徐鹏、马小宁、黄贤俊为本书设计了专栏。

感谢博文视点编辑团队。感谢在本书写作前期与我们合作过的编辑方舟，在写作后期参与合作的编辑徐定翔和李滢波。特别感谢自始至终和作者们一起工作的编辑周筠、杨绣国。他们和作者们一同构思，耐心修改，没有他们的不懈努力，以及细致的编辑和推广工作，就没有《编程之美》的成功上市。

作者

2007 年 11 月

面试杂谈

背景

每年从金秋九月起，校园里的广告栏中、BBS 上的招聘信息就逐渐多了起来。小飞是一名普通高校的应届计算机专业硕士毕业生，他勤奋好学，成绩中上，爱好广泛。看到身边的同学都在准备精美的简历，参加各种各样的招聘会，笔试、面试，他也坐不住了。他在 BBS 上看了各式各样的“面经”，也挤过招聘会上的人潮，长叹：“行路难，行路难，好工作，今安在？”

小飞从网上了解到了有关招聘的各种术语，他整理了一个列表：

名词	解释
面经	面试的经历。
默拒	投了简历，进行了面试，但是公司从此再也没有消息，询问也不回答。
Offer	公司给学生发的入职邀请。
群殴	通常指一群人一起参加面试，一般以多对多的形式同时进行，最后总是会有人被不幸淘汰，这一过程就叫做“群殴”。
听霸	凡校内招聘演讲会都出席旁听的。
投霸	凡公司招人都投简历的。
笔霸	凡投出简历都能得到笔试机会的。
面霸	凡参加笔试都有面试通知的。
巨无霸	在招聘过程屡屡被拒、机会全无的，江湖人称“巨无霸”！
霸王面	“霸王面”指没有获得面试资格，却主动找用人单位，要求面试的人，源自吃饭不给钱的“霸王餐”，即“没机会面，创造机会也要面”。

小飞获得了一个在微软亚洲研究院实习的机会，在工作中认识了一位有丰富招聘经验的研发经理。他对经理进行了非正式的采访，希望能得到第一手的“内幕”消息。下面就是小飞整理出来的问答。小飞的问题用 Q 来标注，经理的回答用 A 标注。

典型面试

备注：在本文中，应聘者（英文为：candidate, interviewee）指应聘公司职位的学生或其他社会人士；面试者（英文为：interviewer）指公司里进行招聘和面试的人员。

Q：经理，您好。我就开门见山，能否分享一下当年您第一次去面试的故事？

A：好，大学毕业后，我进入了学校“产业办”开的公司。有一天，一家美国公司（我们姑且叫它 H 公司）来招人，这是我的第一次面试。那个公司的代表和我寒暄之后，递给我一道题目，题目大意是“写一个函数，返回一个数组中所有元素被第一个元素除的结果”。我当时还问了一些问题，以确保理解无误，所谓 clarification 是也。那位面试者简单地解释了一下，然后就在电脑上敲敲打打，也不理我了。我想这也不难，如何能显示我的功力呢？于是我就把循环倒着写 for (i=n; i>=0; i--)，因为我当时看到一本 UNIX 书上是这么写的。

代码大概是这样的：

```
void DivArray(int * pArray, int size)
{
    for (int i = size-1; i >= 0; i--)
    {
        pArray[i] /= pArray[0];
    }
}
```

写完之后，他看了看就问我，你为什么要这么写循环？如果不这么写可以吗？我说，也可以呀。他问了两遍，如果正着写循环会出现什么问题。我想，能有啥问题？就把循环正着写。噢，原来陷阱在这里！你知道这个陷阱是什么吗？

Q：让我想一想，知道了，如果循环从数组的第一个元素开始，并且不用其他变量的话，在循环的第一步，第一个元素就变成了 1，然后再用它去除以其他元素，就不符合题目要求了。

A：对，同时还有另一个陷阱——看看你是否会检查除数为零的情况，以及对参数的检查，等等。

Q: 这不是很简单吗? 一会儿就写完了。

A: 面试题大多数不难, 但是通过观察应聘者写程序的实际过程, 面试者可以看出应聘者的思维、分析、编程能力。面试者一般还会有后面几招留着。比如, 如果你要测试刚才写的这个函数, 你的测试用例有多少? 或者改变一些条件, 能否做得出来?

Q: 很多人说, 面试是一个不公平的游戏, 因为信息不对称。比如: 面试者知道问题的答案, 而应聘者不知道, 面试者知道今年公司要招几个人, 而应聘者不知道。

A: 但是, 应聘者手头有几个 Offer, 面试者也不知道。应聘者是否喜欢公司提供的职位和薪酬, 面试者也不知道。一方面, 应聘者在“求”职, 另一方面, 面试者也在“求”才。面试也是一个增进双方互相了解的有效途径。

既然扯到了“信息不对称”, 我再讲一个我的故事。当年 H 公司来我校面试的时候, 我对 H 公司的了解仅限于 H 公司捐赠给我们计算机系的一个有些过时的小型机系统。我想, 这个 H 公司是不是还有一些新东西? 那时候还没有互联网, 于是我就托人借了几本原版的 *Byte* 杂志来看, 那是很厚的一本杂志, 非常多的广告, 看了半天, 夹在杂志中的小广告掉了一地。我只看到杂志对 H 公司新出的一个桌面管理软件“NewWave”的评价, 我琢磨了半天, 大概搞懂了这是一个什么东西, 市场上还有什么竞争对手, 等等。

过了两天, 面试开始了, 对方端坐在沙发里问“你对我们 H 公司有何了解?” 我先说了 H 公司的小型机系统, 然后说, “By the way, 我还了解了 NewWave”。于是我把看到的東西复述了一下。没想到对方坐直了身子, 说这个 NewWave 就是他曾经领导的项目。于是我就根据杂志上的描述问, “您怎么看某某竞争产品?” 他很兴奋地跟我谈了 NewWave 是如何的领先, 等等。后来我们又聊了不少相关的东西。

最后所有人面试结束之后, 我们的领导说, 美方觉得我很突出, 知道不少东西, 包括 NewWave, 口语也很好。领导就要求我给所有人都介绍一下 NewWave, 我只好把看到的東西又复述了一次。不久, H 公司过来面试的另一个经理不解地对我们领导说: “为什么你们这么多人知道 NewWave?”

前不久, 我在面试的时候问一位同学, “你对微软亚洲研究院有什么了解?” 他说, “没啥了解, 昨天打电话叫我来面试, 我就来了……” 对于这样的同学, 信息的确是非常不对称, 那他吃亏也是难免的了。还有一位在面试中发挥得很不好的同学跟我说, 他特地没有做任何准备, 因为他想显示他的“raw talent”……

Q: 关于 Test (测试) 的职位, 有没有一些典型的题目呢?

A: 有哇, 典型的题目如给你一支笔, 让你说说你如何测试——据说要测试 12 个方面; 再比如判断一个三角形的特性(直角、钝角、锐角、等腰)——据说有 20 多个测试用例, 这是要考察大家思考问题的全面程度和逻辑分析能力(测试用例见 4.8 节“三角形测试用例”)。

Q: 网上有些非常流行的问题, 都号称是从大公司流传出去的, 是真的吗?

A: 对, 是有一些题目比较常见, 例如“下水道的井盖为什么是圆的”, 还有一个问题一度非常流行, 据说早期应聘 PM (Program Manager 程序经理) 职位的应聘者大多曾碰到这个题目:

房间里有三盏灯, 屋外有三个开关, 分别控制这三盏灯, 只有进入房间, 才能看到哪一个电灯是亮的。请问如何只进入房间一次, 就能指明哪一个开关控制哪一个灯?

传说在晚上, 微软一些会议室的灯忽明忽灭, 就是一些还没有搞懂的同事们在实地钻研。

Q: 我大概了解了 Dev/PM/Test 这三种工作的典型面试题, 那么这些题目的答案别人都知道了, 还怎么面试呢?

A: 对, 会有不少题目流传出去, 这本来无妨。但是一些人知道答案之后, 就开始背诵, 或者原封不动地拿它去面试应聘者, 忘了“知道答案”和“能做一个好员工”的关系。知道了题目的答案, 就能做一个好的开发人员、项目经理, 或者销售经理吗? 一个极端的情况会是: 公司里每一个人都知道哪盏灯是由哪一个开关控制的, 如何测试三角形的类别等, 但是这个公司真能从此开发出更好的软件吗?

一句话: 关键不在于答案, 而在于思考问题的方法, 这也是我们没有“题库”的原因。

研发职位的选择

Q: 微软及很多其他软件公司都有不少研发职位, 名称不尽相同, 而且还是缩写, 能不能讲解一下?

A: 不少同学对微软公司的各种研发职位 (Discipline) 并不太了解, 我们在面试进行到编程之美——微软技术面试心得

一半的时候，经常发现一个应聘者其实更适合做其他类型的工作。当然这时我们可以调换面试的方向，但是对应聘者来说总不是一件好事。我刚好在 BBS 上看到了一篇文章，这篇文章从个人的角度出发，非正式地讲了 R&D 各个方向的特点，虽然并非完全正确，介绍也不一定全面，但是我们不妨看看。

aR: Assistant Researcher, 助理研究员，也可以叫研究员助理，主要在“R&D”的“R”这一端，工作是读论文，提想法，被否决后再提想法（如此反复 N 次），赶在截止时间之前提交论文。aR 的想法得到初步验证之后，还要跟其他部门推销自己的想法，争取把想法变成产品。aR 的乐趣是能在一个领域中深入研究，发表论文，申请专利，每个专利申请（无论是否批准）都能让自己得一块黑色立方体石头（如图 1 所示）。好多人的桌面上堆了不少石头，好像他们没什么苦恼。aR 有时做的事情和 RSDE 差不多。aR 以后会成长为 Associate Researcher（副研究员）、Researcher（研究员）、高级研究员，等等。总之，最后就成了大家小时候特别梦想做的“科学家”。

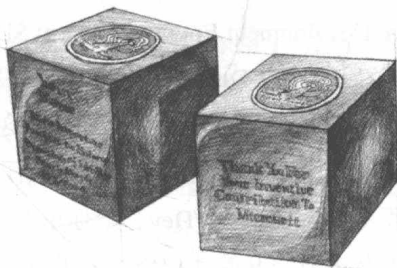


图 1 申请专利得到的石头

Dev: 正式的名称叫 SDE (Software Development Engineer), 这个职位和 aR 相对，是在“R&D”的“D”这一端。他们在一个产品团队中，按照严格的流程开发产品。MS 的一个产品发布之后，所有成员会得到一小块铁皮（学名叫“Ship-it Award”，如图 2 所示），上面写着产品的名字和发布日期，资深的 Dev 会收集到不少，他们会认真地把这些小铁皮整齐地贴起来，摆在办公桌最高的位置上。Dev 的乐不少，这里就不列举了。但是苦也有不少，比如产品的周期有时非常长，过程定义得非常完备（有时不免觉得太完备了）；比如要维护老版本；比如要用比较成熟的技术，而不是用最时髦的东西来开发产品。另外，Dev 要负责一个或几个模块，这些模块不一定和最终用户打交道，未必是整个产品的核心模块。做一个好的 Dev 要生活在代码中，对代码和平台的各种细节要非常熟悉，掌握非常底层的技术，有些人以此为乐，有些人则未必。Dev 的职业发展道路很多，如果只想钻研技术，不乐意做很多