

《编程之道》

蓝森林 <http://www.lslnet.com> 2001年1月1日 21:52

作者: Geoffrey James

[美]Geoffrey James

小赵 录入整理

声明:

本文内容来自《编程之道》一书，小赵只负责录入，所有版权归该书的出版社。本文由原来的三册书组成：《编程之道》《编程之禅》《计算机时代的寓言》

译者序

如今程序设计类的图书可谓是层出不穷，其中绝大部分是以教程、使用手册和编程技巧为主，它们都是用来处理一些技术性的问题的。

然而，怎样领悟程序设计的内涵和精髓呢？程序员编程时是一种怎样的思维方式？他们的工作、生活是个什么样子？他们在想些什么？作为一名经理如何同他们相处并充分发挥他们的积极性？涉及这些方面的书却寥寥无几。但没有并不代表不重要——随着我国程序设计事业从起步到发展，这些方面的内容显得越来越重要，我们越来越希望了解它们。

美国是个软件工业和程序设计水平处于世界依靠地位的国家，我们是不是可以从那里借鉴点什么呢？您将要读到的《编程之道》一书便是出息美国的一位善于进行哲学性思考、有十多年工作经验的程序设计师——Geoffrey James 之手。

在本书中，作者以一种幽默、辛辣的眼光审视着发生在程序设计室里的各种各样的小故事，并运用东方的哲学体系进行深层次的思考和理解，即进行“道”式的思考和理解。简单的故事蕴含深奥的道理，是本书的极大特色。

本书不仅语言优美、比喻生动，而且在结构上颇具匠心。

本书在美国出息时分为三册——《编程之道》《编程之禅》和《计算机寓言——信息时代的启示》，经与作者协商，决定将中译本汇总为一本以《编程之道的》的书名出版。

1、《编程之道》

本书的构思新颖而奇特。作者作为一名计算机考古的业余爱好者，偶然发现了在程序设计业内传说的《编程之道》。这其中的内容，每个程序员都想知道，但谁也没有见到过，就有点像金庸的小说《射雕英雄传》中的《九阴真经》。于是作者便将它“翻译”成此书，公之于众。全书共分为九篇：

第一篇 静寂的空宇

第二篇 古代的大师

第三篇 设计

第四篇 编码

第五篇 维护

第六篇 管理

第七篇 公司里的学问

第八篇 硬件和软件

第九篇 尾声

2、《编程之禅》

在本书中，作者借用中国传统文化中的五行——“金、木、水、火、土”来代表与计算机相关的概念，并以此将多个富有祖宗色彩的故事有机地组织起来。全书分为五篇

第一篇 木

第二篇 火

第三篇 土

第四篇 金

第五篇 水

3、《计算机寓言——信息时代的启示》

作者以四季——“春、夏、秋、冬”来组织本书的结构。全书共分为四篇：

第一篇 秋——虚幻与真实

第二篇 冬——战争与死亡

第三篇 春——真爱与新生

第四篇 夏——启示

我们将此书献给那些爱笑也爱思考的程序设计者，以及那些同程序设计者打交道的管理人员，当然他们也爱笑爱思考。

《编程之道》

The Tao Of Programming —— 编程之道

Translated By Geoffrey James

[[小赵](#)]翻译整理

Book One – The Silent Void

Thus spake the master programmer:

编程大师如是说：

“When you have learned to snatch the error code from the trap frame, it will be time for you to leave.”

“当你从我手中夺走水晶球时，就是你离开的时候了。”

1.1

Something mysterious is formed, born in the silent void. Waiting alone and unmoving, it is at once still and yet in constant motion. It is the source of all programs. I do not know its name, so I will call it the Tao of Programming.

寂静的虚空中诞生了神秘的东西，这种东西恒久存在永不消失，它是所有程序的根源所在，我不知道怎么形容它，姑且称它为编程之道。

If the Tao is great, then the operating system is great. If the operating system is great, then the compiler is great. If the compiler is greater, then the applications is great. The user is pleased and there is harmony in the world.

如果道是完美的，那么操作系统就是完美的，如果操作系统是完美的，那么编译器就是完美的，如果编译器是完美的，那么应用程序就是完美的，所以用户心满意足，整个世界因此和谐。

The Tao of Programming flows far away and returns on the wind of morning.

编程之道去如黄鹤来如晨风。

1.2

The Tao gave birth to machine language. Machine language gave birth to the assembler.

道生机器语言，机器语言生汇编器。

The assembler gave birth to the compiler. Now there are ten thousand languages.

汇编器生编译器，最后产生上万种高级语言。

Each language has its purpose, however humble. Each language expresses the Yin and Yang of software. Each language has its place within the Tao.

不论多么的微不足道，每种语言都有它自己的目的，每种语言都表达了软件的阴阳两极。每种语言都各得其道。

But do not program in COBOL if you can avoid it.

但是尽量不要用 COBOL 语言。

1.3

In the beginning was the Tao. The Tao gave birth to Space and Time. Therefore, Space and Time are the Yin and Yang of programming. 道之初，带来了空间和时间，所以，空间和时间是编程的阴阳两极。

Programmers that do not comprehend the Tao are always running out of time and space for their programs. Programmers that comprehend the Tao always have enough time and space to accomplish their goals.

不懂编程之道的程序员常常把空间和时间消耗殆尽，得道的程序员则总是有足够的空间和时间去完成编程任务。

How could it be otherwise?

否则会是什么样呢？

1.4

The wise programmer is told about the Tao and follows it. The average programmer is told about the Tao and searches for it. The foolish programmer is told about the Tao and laughs at it. 上士闻道，从而行之。中士闻道，谨而寻之。下士闻道，大笑之。

If it were not for laughter, there would be no Tao.

大笑不足为道。

The highest sounds are the hardest to hear. Going forward is a way to retreat. Greater talent shows itself late in life. Even a perfect program still has bugs.

希音不闻，进即是退，大器晚成。任何程序都有漏洞。

Book Two - The Ancient Masters

Thus spake the master programmer:

编程大师如是说：

“After three days without programming, life becomes meaningless.”
三日不编程，食肉无味。

2.1

The programmers of old were mysterious and profound. We cannot fathom their thoughts, so all we do is describe their appearance.

远古时代的编程大师们高深莫测，我们不能揣测他们的所思所想，只能描述外表所见。

Aware, like a fox crossing the water. Alert, like a general on the battlefield. Kind, like a hostess greeting her guests. Simple, like uncarved blocks of wood. Opaque, like black pools in darkened caves.

他达明，如狐狸过水；机警，如战场上的将军；和善，如主妇款待客人；简单，呆若木鸡；混沌，如深渊之水。

Who can tell the secrets of their hearts and minds?

谁能道尽他们的所有？

The answer exists only in the Tao.

答案仅存于道。

2.2

Grand Master Turing once dreamed that he was a machine. When he awoke he exclaimed:

超级大师图灵曾梦见自己是一台机器，醒后他这样回忆：

“I don't know whether I am Turing dreaming that I am a machine, or a machine dreaming that I am Turing!”

“我不知道是图灵梦见自己变成机器还是机器梦见自己变成图灵。”

2.3

A programmer from a very large computer company went to a software conference and then returned to report to his manager, saying: “What sort of programmers work for other companies? They behaved badly and were unconcerned with appearances. Their hair was long and unkempt and their clothes were wrinkled and old. They crashed out hospitality suites and they made rude noises during my presentation.”

一个大公司的程序员参加一个软件会议后向他的主管汇报：“那些别的公司的程序员都是些什么样的人呀？他们举止不雅，不修边幅，头发蓬乱，衣服破旧，根本不热情好客，还在我说话的时候乱嚷嚷。”

The manager said: "I should have never sent you to the conference. Those programmers live beyond the physical world. They consider life absurd, an accidental coincidence. They come and go without knowing limitations. Without a care, they live only for their programs. Why should they bother with social conventions?"

他的主管说：“我不应该让你参加这次会议，这些程序员生活在现实世界之外。他们认为生活是可笑的，一场意外的偶然而已。他们来去自由，无所牵挂，他们只为他们的程序生活。为什么要用世俗的烦扰去扰乱他们呢？”

"They are alive within the Tao."

“他们生活在道中”。

2.4

A novice asked the Master: "Here is a programmer that never designs, documents, or tests his programs. Yet all who know him consider him one of the best programmers in the world. Why is this?"

一个初学者问主管经理：“有一个程序员，他从来不预先设计，也不写文档，甚至不测试他的程序，但是知道他的人都认为他是世界上最伟大的程序员，为什么呢？”

The Master replies: "That programmer has mastered the Tao. He has gone beyond the need for design; he does not become angry when the system crashes, but accepts the universe without concern. He has gone beyond the need for documentation; he no longer cares if anyone else sees his code. He has gone beyond the need for testing; each of his programs are perfect within themselves, serene and elegant, their purpose self-evident. Truly, he has entered the mystery of the Tao."

经理说：“那个程序员掌握了道。他不需要预先进行设计；系统崩溃时他也从不烦躁，只是接受发生的一切而不管发生的事是好是坏。他不需要写文档，他从不顾及有没有人看他写的代码。他也不需要进行测试；他写的每个程序都有一个完美的自我，平静而优雅，它们的目的不言自明。他已经真正掌握了道的精髓。”

Book Three - Design

Thus spake the master programmer:

编程大师如是说：

“When program is being tested, it is too late to make design changes.”

“程序被测试时再去改变它的设计已经太晚了。”

3.1

There once was a man who went to a computer trade show. Each day as he entered, the man told the guard at the door:

曾经有一个人参加计算机展览，每天他进门时都对门卫说：

“I am a great thief, renowned for my feats of shoplifting. Be forewarned, for this trade show shall not escape unplundered.”

“先警告你，我是偷盗高手，我入室偷盗的本领闻名遐迩。这次展览会也再劫难逃。”

This speech disturbed the guard greatly, because there were millions of dollars of computer equipment inside, so he watched the man carefully. But the man merely wandered from booth to booth, humming quietly to himself.

警卫因此大动干戈，因为里面有价值百万的计算机设备，所以他牢牢盯紧了这个人，但是这个人只是从一个摊位逛到另一个摊位，一边平静地喃喃自语。

When the man left, the guard took him aside and searched his clothes, but nothing was to be found.

当这个人离开时，警卫搜了他的身，但是什么也没有发现。

On the next day of the trade show, the man returned and chided the guard saying: “I escaped with a vast booty yesterday, but today will be even better.” So the guard watched him ever more closely, but to no avail.

展览会的第二天，这个人又对警卫说：“昨天我满载而归，但是今天我会收获更大。”所以警卫更加小心地盯紧他，但是这样做完全于事无补。

On the final day of the trade show, the guard could restrain his curiosity no longer. “Sir Thief,” he said, “I am so perplexed, I cannot live in peace. Please enlighten me. What is it that you are stealing?”

展览会的最后一天，警卫再也不能忍住他的好奇心了。“小偷先生，”他说，“你说我惊慌不安，请告诉我，你到底偷了什么？”

The man smiled. “I am stealing ideas,” he said.

这个人笑笑说：“我在偷想法。”

3.2

There once was a master programmer who wrote unstructured programs. A novice programmer, seeking to imitate him, also began to write unstructured programs. When the novice asked the master to evaluate his progress, the master criticized him for writing unstructured programs, saying: "What is appropriate for the master is not appropriate for the novice. You must understand the Tao before transcending structure."

有一位编程大师，他写非结构化的程序，一位初学者刻意模仿他，也写非结构化的程序。当他让大师看他的进步时，大师批评了他的非结构化程序：“对一位编程大师合适的东西未必对一个初学者同样合适，在超越结构化之前，你必须理解编程之道。”

3.3

There was once a programmer who was attached to the court of the warlord of Wu. The warlord asked the programmer: "Which is easier to design: an accounting package or an operating system?"

有人问一位程序员，“一个财务软件和一个操作系统哪个更容易设计？”

"An operating system," replied the programmer.

“是操作系统”，这位程序员回答说。

The warlord uttered an exclamation of disbelief. "Surely an accounting package is trivial next to the complexity of an operating system," he said.

此人大惑不解。他说：“显然一个财务软件比起操作系统来说其复杂性是微不足道的”。

"Not so," said the programmer, "when designing an accounting package, the programmer operates as a mediator between people having different ideas: how it must operate, how its reports must appear, and how it must conform to the tax laws. By contrast, an operating system is not limited by outside appearances. When designing an operating system, the programmer seeks the simplest harmony between machine and ideas. This is why an operating system is easier to design."

程序员说：“不，设计财务软件时，一个程序员必须成为持不同意见的用户与计算机的一个中介，他必须了解用户的操作习惯，报表要是什么形式，如何遵循税法。相反，一个操作系统完全与这些外部的东西无关。设计操作系统，程序员只需要达到自己的设想与机器之间的简单的和谐。这就是为什么操作系统反而比财务软件更容易设计。”

The warlord of Wu nodded and smiled. "That is all good and well, but which is easier to debug?"

这些人笑着说。“不错，但是哪一个更容易调试呢？”

The programmer made no reply.

程序员没有回答。

3.4

A manager went to the master programmer and showed him the requirements document for a new application. The manager asked the master: "How long will it take to design this system if I assign five programmers to it?" 一个项目经理带给编程大师一个项目的需求，然后问大师：“如果我给你 5 个程序员，要多少时间设计这个项目？”

"It will take one year," said the master promptly.

“一年”，大师说。

"But we need this system immediately or even sooner! How long will it take it I assign ten programmers to it?"

“但是我们等不了那么长时间，越快越好，如果 10 程序员呢？”

The master programmer frowned. "In that case, it will take two years."

大师皱了一下眉头说：“那就要花 2 年”。

"And what if I assign a hundred programmers to it?"

“那，100 程序员呢？”

The master programmer shrugged. "Then the design will never be completed," he said.

大师耸了耸肩说：“那这个项目就永远完不成了。”

Book Four – Coding

Thus spake the master programmer:

编程大师如是说：

"A well-written program is its own heaven;
a poorly-written program is its own hell."

“写的好的程序是它自己的天堂，写的不好的程序是它自己的地狱”

4.1

A program should be light and agile, its subroutines connected like a strings of pearls. The spirit and intent of the program should be retained throughout. There should be neither too little nor too much, neither needless loops nor useless variables, neither lack of structure nor overwhelming rigidity.

一个程序应该是轻灵自由的，它的子过程就象串在一根线上的珍珠。自始至终，目标明确。增之一字节则太长，减之一字节则太短。既没有不必要的循环也没有没有被引用的变量，既不缺少结构化也不至于僵硬呆板。

A program should follow the *Law of Least Astonishment*. What is this law? It is simply that the program should always respond to the user in the way that astonishes him least.

一个程序应该遵循‘`最小惊讶原则’’，什么是‘`最小惊讶原则’’？就是说一个程序应该最小程度地引起用户的惊讶。

A program, no matter how complex, should act as a single unit. The program should be directed by the logic within rather than by outward appearances.

无论一个程序无论有多么复杂都应该是一个单一的整体。程序是被它的内部逻辑所指引而不是它的外观表现。

If the program fails in these requirements, it will be in a state of disorder and confusion. The only way to correct this is to rewrite the program.

如果一个程序不能满足所要的需求，它就会处于混乱无序的状态中。唯一的出路就是重写这个程序。

4.2

A novice asked the master: "I have a program that sometimes runs and sometimes aborts. I have followed the rules of programming, yet I am totally baffled. What is the reason for this?"

一个初学者问大师：“我的一个程序有时正常有时不正常。我已经完全遵循编程的规则，而且我完全被它弄糊涂了，到底是什么原因会导致这样呢？”

The master replied: "You are confused because you do not understand the Tao. Only a fool expects rational behavior from his fellow humans. Why

do you expect it from a machine that humans have constructed? Computers simulate determinism; only the Tao is perfect.

大师说：“你困惑是因为你不了解道。只有傻瓜才会相信只要遵循别人所说就能得到想当然的结果。为什么你要从一个人自己构造的机器中去得到想当然的结果呢？计算机只是决定论的产物；只有道才是唯一完美的。

The rules of programming are transitory; only the Tao is eternal. Therefore you must contemplate the Tao before you receive enlightenment.”

任何编程的规则都只适合于特定的情况，只有道才是永恒不变的。所以在你受到指引之前要先去思考道。”

“But how will I know when I have received enlightenment?” asked the novice.

“但是我怎么知道我何时受到指引呢？”

“Your program will then run correctly,” replied the master.

“当你的程序正确运行时”。大师说。

4.3

A master was explaining the nature of the Tao to one of his novices, “The Tao is embodied in all software -- regardless of how insignificant,” said the master.

初学者请大师解释“道法自然”。大师说：“道存在于任何软件中——无论是多么没有意义的软件”。

“Is the Tao in a hand-held calculator?” asked the novice.

“难道也存在于手持式的计算器里？”，初学者问。

“It is,” came the reply.

“是的，”

“Is the Tao in a video game?” continued the novice.

“也在游戏机里？”，初学者又问到。

“It is even in a video game,” said the master.

“是的，甚至也存在于游戏机里”。大师说。

“And is the Tao in the DOS for a personal computer?”

“也存在于个人计算机的 DOS 里吗？”

The master coughed and shifted his position slightly. "The lesson is over for today," he said.

大师咳嗽了一声，稍微动了一下，说，“今天的课就到这里”。

4.4

Price Wang's programmer was coding software. His fingers danced upon the keyboard. The program compiled without an error message, and the program ran like a gentle wind.

###编程大师编写软件时，手指在键盘上快速飞舞。程序编译时没有一条错误信息，程序运行起来就象一阵微风吹过。

Excellent!" the Price exclaimed, "Your technique is faultless!"

太精彩了！你的技巧已经无可挑剔了。

"Technique?" said the programmer, turning from his terminal, "What I follow is the Tao -- beyond all technique. When I first began to program I would see before me the whole program in one mass. After three years I no longer saw this mass. Instead, I used subroutines. But now I see nothing. My whole being exists in a formless void. My senses are idle. My spirit, free to work without a plan, follows its own instinct. In short, my program writes itself. True, sometimes there are difficult problems. I see them coming, I slow down, I watch silently. Then I change a single line of code and the difficulties vanish like puffs of idle smoke. I then compile the program. I sit still and let the joy of the work fill my being. I close my eyes for a moment and then log off."

“技巧？”，大师转过身说，“我所遵循的是道——它超乎所有的技巧。当我开始编程时我看到的是整个一大块的程序，三年后我看到的是子过程。现在我什么也看不到了。我的整个存在是没有任何形式的虚无。我感觉很悠闲，总之，事实上是我的程序自己在写，有时我看到一些问题，我看到它们，就停下来静静地观察它们，然后我改变了一行代码，难题就象一阵轻烟一样化为乌有。然后我编译程序。坐在那里享受工作的喜悦。闭了一会眼睛然后退出系统。

Price Wang said, "Would that all of my programmers were as wise!"

“什么时候我的程序员才能都达到这样的境界！”

Book Five - Maintenance

Thus spake the master programmer:

编程大师如是说:

"Though a program be but three lines long, someday it will have to be maintained."

“即使一个程序只有三行长，也总有一天需要去维护它”

5.1

A well-used door needs no oil on its hinges.

A swift-flowing stream does not grow stagnant.

Neither sound nor thoughts can travel through a vacuum.

Software rots if not used.

经常使用的门不需要往门轴里上油。

流动的东西就不会停滞不前。

流水不腐，户枢不蠹。——王磊

声音和思想都不能在真空中传播。

软件不用就会腐朽。

These are great mysteries.

5.2

A manager asked a programmer how long it would take him to finish the program on which he was working. "I will be finished tomorrow," the programmer promptly replied.

经理问程序员要多长时间才能完成他的项目。“明天”

"I think you are being unrealistic," said the manager. "Truthfully, how long will it take?"

“太不着边际了，老实说，要多久？”

The programmer thought for a moment. "I have some features that I wish to add. This will take at least two weeks," he finally said.

程序员想了一想说。“还有一些新的功能要加进去，可能至少要两个星期吧。”

"Even that is too much to expect," insisted the manager, "I will be satisfied if you simply tell me when the program is complete."

“即使两个星期恐怕也太夸张了，什么时候你只要告诉我说程序已经完成就好了。”

The programmer agreed to this.

Several years slated, the manager retired. On the way to his retirement lunch, he discovered the programmer asleep at his terminal. He had been programming all night.

几年后，这个经理已经退休了，在他的离职午餐上，他发现这些程序员在他的终端前睡着了，他整夜都在编程。

5.3

A novice programmer was once assigned to code a simple financial package. 一个初学者被要求编写一个财务软件。

The novice worked furiously for many days, but when his master reviewed his program, he discovered that it contained a screen editor, a set of generalized graphics routines, and artificial intelligence interface, but not the slightest mention of anything financial.

他疯狂地工作了很多天，但他的主管检视他的程序时发现，它写了一个编辑器，一个图形程序集，和人工智能的界面，但是看不到任何跟财务有关的东西。

When the master asked about this, the novice became indignant. "Don't be so impatient," he said, "I'll put the financial stuff in eventually."

主管要求解释时，程序员被激怒了：“你太没耐心了，我会在最后写财务的部分。”

5.4

Does a good farmer neglect a crop he has planted?
Does a good teacher overlook even the most humble student?
Does a good father allow a single child to starve?
Does a good programmer refuse to maintain his code?

一个好的农民不会不管他的庄稼。
一个好的老师不会不管哪怕是最差的学生。
一个好的父亲不会让他的任何一个孩子挨饿。
一个好的程序员不应拒绝维护他的程序。

Book Six - Management

Thus spake the master programmer:

编程大师如是说：

"Let the programmer be many and the managers few -- then all will be productive."

“增加程序员，减少经理——这样他们都可提高效率。”

6.1

When managers hold endless meetings, the programmers write games. When accountants talk of quarterly profits, the development budget is about to be cut. When senior scientists talk blue sky, the clouds are about to roll in.

经理们忙于无休无止的会议时，程序员们在写游戏。财务主管在谈论季度盈利时开发预算将会被削减。当老科学家在谈天空的蓝色时，云团蜂拥而至。

Truly, this is not the Tao of Programming.

这不是编程之道。

When managers make commitments, game programs are ignored. When accountants make long-range plans, harmony and order are about to be restored. When senior scientists address the problems at hand, the problems will soon be solved.

经理分配任务时不会包括游戏，会计师做长期计划时，应该做到面面俱到，皆大欢喜。科学家提出一个问题，就会很快得到解决。

Truly, this is the Tao of Programming.

这才是编程之道。

6.2

Why are programmers non-productive? Because their time is wasted in meetings.

为什么程序员没有效率，因为他们把时间都浪费在开会上了。

Why are programmers rebellious? Because the management interferes too much.

为什么程序员难于管理？因为管理者的干预太多了。

Why are the programmers resigning one by one? Because they are burnt out.

为什么程序员一个接一个地辞职，因为他们累坏了。

Having worked for poor management, they no longer value their jobs.

在糟糕的管理下工作，他们享受不到工作的乐趣。

6.3

A manager was about to be fired, but a programmer who worked for him invented a new program that became popular and sold well. As a result, the manager retained his job.

一个经理将被解雇，但他手下的一个程序员写了一个流行的软件并且销路很旺。所以经理保住了他的工作。

The manager tried to give the programmer a bonus, but the programmer refused it, saying, "I wrote the program because I thought it was an interesting concept, and thus I expect no reward."

经理为程序员分红，但程序员拒绝了，他说：“我写程序是因为写程序很有趣，所以我并没想过得资金”。

The manager, upon hearing this, remarked, "This programmer, though he holds a position of small esteem, understands well the proper duty of an employee. Let's promote him to the exalted position of management consultant!"

这位经理说，“这位程序员，不管他现在是什么位置，他都能理解一个雇员的职责所在。我们要把他提升为管理顾问”。

But when told this, the programmer once more refused, saying, "I exist so that I can program. If I were promoted, I would do nothing but waste everyone's time. Can I go now? I have a program that I'm working on."

这位程序员又一次拒绝了，他说：“我是一个程序员所以我可以编程，如果我被提升了，除了浪费别人的时间什么也做不了，现在我可以走了吗？我还有一个程序没写完。”

6.4

A manager went to his programmers and told them: "As regards to your work hours: you are going to have to come in at nine in the morning and leave at five in the afternoon." At this, all of them became angry and several resigned on the spot.

经理对程序员说，“你们的工作时间是早上9点到正午点。”，所有的程序员都很不满。

So the manager said: "All right, in that case you may set your own working hours, as long as you finish your projects on schedule." The programmers,

now satisfied, began to come in a noon and work to the wee hours of the morning.

经理又说：“好吧，那随你们的便，只要能按时完成任务。”，程序员们这下满意了，他们中午上班，一直工作到凌晨。

Book Seven – Corporate Wisdom

Thus spake the master programmer:

编程大师如是说：

“You can demonstrate a program for a corporate executive, but you can't make him computer literate.”

###

“你可以向一位公司主管演示一套程序，但你不能让他的电脑变得象人一样。”

7.1

A novice asked the master: “In the east there is a great tree-structure that men call 'Corporate Headquarters'. It is bloated out of shape with vice-presidents and accountants. It issues a multitude of memos, each saying 'Go, Hence!' or 'Go, Hither!' and nobody knows what is meant. Every year new names are put onto the branches, but all to no avail. How can such an unnatural entity exist?”

一个新学问大师：“在东方，有一个庞大的机构，人们称作总部。它由为数众多的大小领导。每天发出大量的备忘录，每个备忘录都说：‘干这个，干那个’，没有一个人知道它是什么意思。每年这个机构都会有更多的人加进来而变得越来越大，但是去没有人去做事情。怎么烩样奇怪的东西还能存在呢？”

The master replies: “You perceive this immense structure and are disturbed that it has no rational purpose. Can you not take amusement from its endless gyrations? Do you not enjoy the untroubled ease of programming beneath its sheltering branches? Why are you bothered by its uselessness?”

大师说：“你已经知道这个巨大的机构并不合理，你就不能不管它为什么这样吗？你尽管享受它的好处就是了，干吗要去管它到底是没有效率呢？”

7.2

In the east there is a shark which is larger than all other fish. It changes into a bird whose wings are like clouds filling the sky. When this bird moves across the land, it brings a message from Corporate Headquarters. This message it drops into the midst of the programmers, like a seagull

making its mark upon the beach. Then the bird mounts on the wind and, with the blue sky at its back, returns home.

在遥远的东方，传说有一只巨大的鱼，叫鲲，它变成了一只大鸟，叫鹏，鹏的翅膀可以遮天蔽日，这只大鸟经过陆地的时候，它带来了一个消息，它把这个消息丢在程序员中，就象一只海鸥在海滩上做的记号，然后鹏背负蓝天，乘风而上，回家去了。

The novice programmer stares in wonder at the bird, for he understands it not. The average programmer dreads the coming of the bird, for he fears its message. The master programmer continues to work at his terminal, for he does not know that the bird has come and gone.

初学者惊恐地看着这只鸟，根本不敢相信，中级程序员担心这只鸟的到来，因为它害怕它带来的消息，编程大师则继续在他的终端前工作他根本不知道这只鸟曾经来过。

7.3

The Magician of the Ivory Tower brought his latest invention for the master programmer to examine. The magician wheeled a large black box into the master's office while the master waited in silence.

一个学究带着他的最新发明去见编程大师，他把一个大黑盒子推进大师的办公室。

"This is an integrated, distributed, general-purpose workstation," began the magician, "ergonomically designed with a proprietary operating system, sixth generation languages, and multiple state of the art user interfaces. It took my assistants several hundred man years to construct. Is it not amazing?"

“这是综合的、分布式的通用工作站，”，学究开始介绍了，“按人体工学设计的操作系统，第六代的高级语言，多样的用户界面，整整花了我的助手好几百人年的时间，挺不错吧？”

The master raised his eyebrows slightly. "It is indeed amazing," he said.

大师扬了一下眉毛，说：“的确很了不起。”

"Corporate Headquarters has commanded," continued the magician, "that everyone use this workstation as a platform for new programs. Do you agree to this?"

“头儿已经说了，”这位学究继续说，“每个人都要用这个工作站作平台，你觉得怎么样？”

"Certainly," replied the master, "I will have it transported to the data center immediately!" And the magician returned to his tower, well pleased.

“当然了，”，大师回答说，“我要马上把它送到数据中心去！”，这位学究满意去回去了。

Several days later, a novice wandered into the office of the master programmer and said, "I cannot find the listing for my new program. Do you know where it might be?"

几天后，一个新手走进大师的办公室说，“我找不到我新写的程序了，你知道它可能会在哪儿吗？”

"Yes," replied the master, "the listings are stacked on the platform in the data center."

“当然，”，大师说，“你的程序在数据中心的平台里放着。”

7.4

The master programmer moves from program to program without fear. No change in management can harm him. He will not be fired, even if the project is canceled. Why is this? He is filled with the Tao.

编程大师从一个项目转到另一个项目而毫无顾虑，任何管理上的变动都不能影响他，他不会被解雇，即使项目被取消了。为什么呢？他了解了道。

Book Eight – Hardware and Software

Thus spake the master programmer:

"Without the wind, the grass does not move.

Without software, hardware is useless."

风不动则草不动，

没有软件，硬件只是一堆发热的电子器件

8.1

A novice asked the master: "I perceive that one computer company is much larger than all others. It towers above its competition like a giant among dwarfs. Any one of its divisions could comprise an entire business. Why is this so?"

一个初学者问编程大师：“我知道有一家计算机公司比所有其它的公司都大。在它面前它的竞争对手就象站在巨人面前的侏儒。它的任一部门都有各自的业务，为什么会这样呢？”

The master replied, "Why do you ask such foolish questions? That company is large because it is so large. If it only made hardware, nobody would buy it. If it only maintained systems, people would treat it like a servant. But because it combines all of these things, people think it one of the gods! By not seeking to strive, it conquers without effort."

大师说，“为什么你问如此愚蠢的问题呢？那家公司大是因为它就是如此之大，如果它只做硬件，就没有人会买，如果它只维护系统，人们只会把它当作仆人，但是它的业务囊括了所有这些东西，人们就对它顶礼莫拜好象不用努力奋斗就可击败竞争对手

8.2

A master programmer passed a novice programmer one day. The master noted the novice's preoccupation with a hand-held computer game. "Excuse me", he said, "may I examine it?"

有一天编程大师碰见了一个正玩掌上游戏机的编程新手，他说：“给我看一下？”

The novice bolted to attention and handed the device to the master. "I see that the device claims to have three levels of play: Easy, Medium, and Hard", said the master. "Yet every such device has another level of play, where the device seeks not to conquer the human, nor to be conquered by the human."

新手把游戏机递给大师，“我看到上面说明了三种难度的玩法：容易，中等，高难”，大师说，“但是每个这样的游戏机都有另外一种玩法，此时游戏机即不是想着怎么赢你，也不是怎么被你赢。”

"Pray, great master," implored the novice, "how does one find this mysterious setting?"

“那么请问大师”，初学者说，“这种玩法到底是怎样玩的呢？”

The master dropped the device to the ground and crushed it under foot. And suddenly the novice was enlightened.

大师把游戏机摔到地上然后用脚踩碎它，突然初学者明白了。

8.3

There was once a programmer who worked upon microprocessors. "Look at how well off I am here," he said to a mainframe programmer who came to visit, "I have my own operating system and file storage device. I do not have to share my resources with anyone. The software is self-consistent and easy-to-use. Why do you not quit your present job and join me here?" 有一个工作在微机上的程序员对前来参观的大型机程序员说：“看看我这里吧，我有自己的操作系统和文件存储设备，我不需要与别人共享任何资源，软件条条有理并且易于操作。为什么你不辞去你的工作到这里来呢？”

The mainframe programmer then began to describe his system to his friend, saying: "The mainframe sits like an ancient sage meditating in the midst of the data center. Its disk drives lie end-to-end like a great ocean of machinery. The software is a multi-faceted as a diamond and as convoluted as a primeval jungle. The programs, each unique, move through the system like a swift-flowing river. That is why I am happy where I am."

大型机的程序员对他的朋友说：“大型机就象古代圣贤一样深思熟虑，它的磁盘设备从头到尾就象是一个巨大的机械设备的海洋。软件象钻石一样有多个面，象远古的森林一样令人费解。每一个程序都象水经过河流一样穿行于系统中，那就是为什么我乐于为它工作”

The microcomputer programmer, upon hearing this, fell silent. But the two programmers remained friends until the end of their days.

微机程序员听了这些话后，默默不语，但这两个程序员至死都是很好的朋友

8.4

Hardware met Software on the road to Changtse. Software said: "You are the Yin and I am the Yang. If we travel together we will become famous and earn vast sums of money." And so the pair set forth together, thinking to conquer the world.

硬件和软件在路上碰见了，软件说：“你是阴我是阳，如果我们联手就可以赚大钱。”所以它们走到了一直，考虑怎样征服世界。

Presently, they met Firmware, who was dressed in tattered rags, and hobbled along propped on a thorny stick. Firmware said to them: "The Tao lies beyond Yin and Yang. It is silent and still as a pool of water. It does not seek fame, therefore nobody knows its presence. It does not seeks fortune, for it is complete within itself. It exists beyond space and time."

不久，它们遇到了固件，固件衣衫褴褛，拄着一根拐棍步履蹒跚。固件对它们说：“道在于阴阳之外。它静如止水，它从不指望能扬名远近，所以没有人知道它的

存在。它也不寻求幸福，它只是与它自己的存在在一起，它存在于空间和时间之外。”

Software and Hardware, ashamed, returned to their homes.

软件和硬件面有惭色，各自回了家。

Book Nine - Epilogue

Thus spake the master programmer: 编程大师如是说:

“Time for you to leave.”

“现在是你出师的时候了。”

《编程之禅》

作者: Geoffrey James

小赵 录入整理

前言

《编程之道》的出版在程序设计业内得到了普遍接受，出版社邀请我翻译一些与之相关的文章，以作为那部著名的经典之作的补充。虽然，我申明我的能力难以胜任，但我最终还是被说服试一试。

本书是许多个月来研究和翻译的结果，它试图通过对那些传统著作的摘录，来涵盖一个复杂的主题。我不怀疑，将会有许多计算机考古专家对我的摘录标准提出意见。他们地问，“他为什么不把《UNIX 程序员、大象和妓女》的寓言包括进来？”“他竟敢忽视悠久的历史故事《图灵集市历险记》！”对于这些批评，我只能说，我已经尽了我最大的努力来选择有代表性的内容。

为了确定文中各个章节的年代，我利用了朝代系统。对于那些不熟悉这套划分年代的方法的人来说，可以按四个朝代（或者说“四代”）来划分。

第一个朝代，即所谓的“黄金时代”，要追溯到由玻璃管构建计算机的日子；许多现代的学者断言，这个年代具有神话色彩。第二个朝代开始于晶体管的发明，而结束于集成电路的再现。

现代计算机的历史，开始于第三个朝代，它由主机和控制主机的军机大臣所统治。第四个朝代开始于对“集成教”的镇压，“集成教”对业已建立的秩序的反抗，被狂热的“蓝色兵团”残酷地摧毁了。具有讽刺意义的是，正是这次镇压，导致编程之禅扩散到了外部世界。

除了组成该书主体的古老的素材以外，我还有幸得到了 Babbage 博士和 Yu 博士的帮助，他们分别为该书撰写了引言和序。我希望他们对该书的贡献能够或多或少地弥补我作为编者的不才。

Geoffrey James

1988 年 1 月于洛杉矶

目录

前言

序

引言

第一篇 木

第二篇 火

第三篇 土

第四篇 金

第五篇 水

根据古老的“五行学说”，本书由五部分：

木 火 土 金 水

大师：忍者 行者 隐者 大力神 阿幸

媒体：编年史 民间故事说教 公案 俳句

硬件：键盘 显示器 CPU 打印机 磁盘

方向：东 南 中 西 北

软件：编辑程序 格式化程序 调试程序 解释程序 编译程序

感觉：触觉 视觉 嗅觉 听觉 味觉

阶段：界面 设计 编码 排错 测试

动物：龙 凤凰 牛 兔 蛇

系统：VMS CP-6 OS/VS MS-DOS Unix

这套分类系统包罗万象，要想在该书的篇幅内把它阐释清楚是不大可能的。然而，在沉思于这些千变万化的对应关系之后，您或许会发现其中那令人惊异的洞察力。

序

毫无疑问，古老的编程艺术在西文人的脑海里通常会被误解，现代的观点则认为这是一种工程学、机械论和唯物论。

许多人认为，编程仅是为了达到一个目的而使用的一种手段，而且，一个程序（及程序员）只能依据其赚钱的能力来评价。

这些原始的错觉，来自于对编程的真正意图的一种很深的误解。

高超的程序员并非致力于表面的成功，而是在人机之间寻找一种神秘的存在。

根据禅的解释，在硬件、软件、固件、界面和理解之间没有分界——相反，所有这些都结合进了一个和谐的整体。

只有当程序员最终学会了将那种“自我自我的存在”的虚假感觉抛在一边，此种境界才可能达到，而那种“自我的存在”的感觉往往伴随在我们大多数人的生活之中。

这是计算机所带来的一一禅师程序员的行为方式。

有人说，掌握了禅机的程序员也就掌握了生命。这样的程序员是以一种永不疲倦的童稚的欢乐来看待这个世界的。

觉悟了的程序员，走在大街上也能感觉到两旁房子和大楼里面的电脑。

觉悟了的程序员，能够感觉和听到调制过的数据穿过电脑线时那种电流脉冲的持续不断的嗡嗡声。

觉悟了的程序员已经与宇宙合为一体。

我以前的学生 Geoffrey 能够完成这样一个工作，将失落的编程之禅的经典著作再现于世，作为一名教师，我感到由衷的满足。

真心希望，这本书能够重新确立禅在完善的程序员教育中的重要地位。

C. P. Yu 博士

莲花大学超越机器学院

西藏拉萨 引言

当 James 先生要求我给他的这本书写一篇引言的时候，除了讲述一下关于我自己在程序维护这个神秘的领域中的个人经历，我实在想不到更好的办法了。程序的维护是程序设计艺术中很少被人理解的领域。

有的读者肯定会坚持说，在禅的深奥教义和程序维护的卑微技艺之间，并没有什么共同点。但正如一位大师所言：“方法和途径（也就是道）存在于所有的程序之中，甚至在电子游戏里。”因此，长期被的程序维护艺术必定有其禅的方面，这点是确凿无疑的，虽然那对于未经训练的头脑来说或许不会立刻显现出来。

我的故事开始于我从大学计算机专业毕业后的几个星期。我大学毕业后的目标是为一家研究和开发机构工作，最好是编译程序或操作系统设计的部门。我最后终于找到一家愿意盲雇佣我的机构，但条件是我必须在一段时间内，通过完成程序维护的工作来“学会这套系统”。

我当然会对这个建议产生抵触心理。我花了五年的大学时间，竟然是为了浪费时间来解决其他一些程序员的错误！然而，因为他们曾经答应日后让我干感兴趣的工作，于是我便接受了，并在心中提醒自己，即使这份工作干不出名堂，我还能找到其他的工作。

当我第二个礼拜去报到的时候，我被领着去与那个程序维护组的师父见面。人事部总管带着我快步地穿过开发中心黑暗的走廊，最后她指着长长的大厅通道尽头处的一扇门说：“他在那里面。”说完志身走了，好像有些心神不定。

我向门口走去，眼睛朝里面窥探，我看见一个男人正坐在终端前工作，但他背对着我，所以我不知道他有多大年纪，长得什么样。我只有靠咳嗽一声来表明我的存在，这里，那位大师甚至没有回头瞅一眼便说：“请坐”。

越过他的肩膀，我瞥见了那些难以理解的屏幕显示，随着他纤长的手指在键盘上飞舞，它们一闪一闪地出现在他的终端上。最后，他满意地咕哝了一声，退出系统，然后转过身来面对着我。

我所看见的让我吃了一惊，因为他看上去不像那种应该是禅师的人，他的脸乏味，几乎是丑陋的，他的头发如同一圈烦恼的光环。但人们首先注意的是他的眼睛，透过厚厚的眼镜片射出淡淡的蓝光。

他从头到脚地打量了我一番，然后点点头，就好像确认一个个人观点，“你就是那个新来的？”他酸酸地问。

“是的。”我回答。我假装充满激情，把我的经历和在大学的成绩向他作了快速的汇报。

这位礼貌的听着，然后说：“那很好，但你以前做过程序的维护工作吗？”

我坦白地告诉他我没有做过。

师父长长地叹了口气，“那好，我们应该做我们做做的事。”他说。然后他从一个架子上取下一本庞大的程序清单，随意地翻开它，并递给我。接着他问：“你怎样对付这个？”

我眼睛盯着这本清单，那上面是汇编语言，还掺杂着一些奇怪的宏语言。每隔十几行语句，控制权便转移到某个神秘的子程序，而且，即使这个程序具有任何结构性，我也看不出来。“这是什么程序、”我问。

师父把那本清单从我的膝盖上拿了过去，“这是《世代大师编码藏经》，”他说，“当你学会了从迷魂阵中把错误的代码抓出来时，那就是你出师的时候了。”然后，他合上清单，把它放回到书架上。

我不久便意识到，程序的维护远比我想象的要难得多。我起初试图学会那本《编码藏经》中所用的汇编语言，但令我烦恼的是，我发现那套汇编语言从未按规矩提供过说明文档，现存的仅有一些笔记，是一位多年前就死去或离开公司的硬件开发者写的。

这本《编码藏经》不能给我什么帮助了。虽然其中偶尔有一些注释，但这些注释和这种汇编语言一样模糊，除了一些涉及到有关原始硬件构造的使人干着急的资料以外，别无它物。

当我向那位师父抱怨这一切的时候，他礼貌地听着，并在我们两人沉默了许久之后，回答我说：

“你正在试图理解某种不可能被你的理性思维所理解的东西，”他说，“这样所导致的结果必然是失败。你必须先清空你的大脑，只有那时，你才能开始领悟《编码藏经》。”

接着，师父开始慢慢地向我阐释那本《编码藏经》里复杂的逻辑。当我聆听他那温和的声音时，我终于开始觉察到一点光芒的闪烁，这是隐藏在《编码藏经》中辉煌的和永恒的灵光。“那些大师对‘良好的程序设计惯例与规则’一无所知，”师父说，他们努力去理解普遍意义上的计算机思维的内部活动，他们还需要什么说明文档呢？那些程序是极限的表达。

然而，即使我慢慢开始有所领会，我还是感觉到自己像一只在琥珀在挣扎的小飞虫。师父所讲的与我曾学过的东西截然不同，我的理性思维很难接受。但师父总是耐心地一遍遍解释说，我必须不靠理性思维去推理，而要用潜意识来领悟《编码藏经》的内涵。

经过数月的指点后，我感到很自信了，便想试试我的第一块修补程序。为了给师父一个惊喜，我偷偷地工作。我写了一段能重新运行几行语句的补丁，把程序重新进行汇编，并把新程序释放到生产系统中去。

第二天早上，我来晚了一会儿。让我感到奇怪的是，开发中心的主任和从事部总管正在师父的办公室里。当我走进大厅后，从事部总管看见了我就把门关上了。我听见了很大的说话声，但听不清说什么。

等到那两位来访者离开后，我走进师父的办公室，“什么事？”我问。

“你的修补程序，昨天晚上六点整进入了生产系统。它现在已经被删除了。”

“那么？”

“你仍然可以继续工作。”师父说。

最后，我终于明白，所有用我的理性思维去理解《编码藏经》的努力都是徒劳的，这使我很绝望。师父觉察到了我思想上的这个变化，他开始向我传授开机。他教给我冥想和查错的技巧，他说这些技巧是从计算机时代之初，由技术支持部一代一代地传下来的。

我听着听着，开始意识到一个关于我以前的编程体验的特大事实。在大学里，我一起认为程序员的工作主要是控制硬件和软件的活动，编程的最高艺术境界是，为了完成一项任务或达到一个目标而成功地运用高超的编程技巧。但程序维护是不同于程序开发的。要想维护一个程序，就要把程序看作是一棵渐渐成长的植物，拔苗助长是毫无益处的。事实上，这样的行为更容易导致植物的死亡。一个程序必须被小心地滋养。程序员在对程序进行改动之前，必须对每一个逻辑关系非常熟悉，必须对程序的意图有很深刻的理解力。这种理解力不是在一夜之间就可以得来的，需要长时间的积累。

好几个月以后，我终于能够给《编码藏经》成功地设计修补程序了，但设计之前要经过长时间的冥想，还要把那本程序清单打开着支在我的桌子上。我还发现，如果我工作的时候点燃一支香，口中不停在重复师父教给我的口诀：

“Null-So-Stix-Etx-Eot”（ASCII 码表中头五个字节 NUL、SOH、STX、ETX、EOT 的发音——译注），我更容易集中注意力。师父说，那个口诀意味着宇宙的“五原”。

不久我发现我不再关心是否能为工作成绩而得到奖励了，在我自身和我所维护的程序之间也看不见任何分离了。我像一个一生都活在阴影里的人一样，开始理解编程之禅，这是一种隐于程序员设计背后的难以言喻和难以形容的力量，就像投射影子的太阳一样。

从那毫无意义的“自我的存在”中摆脱出来后，我开始发觉，那些伟大的编程语句过去对我来说似乎模糊不清，只是因为我还没有彻悟，无法理解它们。我现在知道了为什么过去的那些程序员从不为他们的程序提供文档，因为英语语言的描述本身更让人迷惑，而非使人受到启发。

一天，我发现自己正在解决的一个问题涉及到《编码藏经》中最复杂的部分——错误分析全程。我在对此一无所知的情况下，给出了一个修补程序，它能通过对硬件中断区的内容进行检查来测定错误的情况，使程序能继续正确执行。

那天下午，师父第一次走进了我的工作间，他把手放在我的肩膀上，眼睛朝下看着我，“现在该是你出师的时候了。”他说。

这些就是我第一次接触禅式程序设计的经历。虽然从那以后我被指派参加过很多项目设计，但我从未忘记过我的第一位师父给我的教诲。

想象一下我的惊奇——当我在《编程之禅》中发现了如此之多的我师父非常喜爱的语言时。我终于看到了那些隐于他那难忘的演说背后的古代传统。

James 先生将那本经典的影响深远的著作重新发掘出来，我们大家都欠他一份人情。要不是他的持之以恒，它可能会永远失传。在本书中，James 先生收集了一大堆定期，如异端的说教、民间故事和诗歌等，它们组成了具有传奇色彩的“集成教”的教义。正是通过 James 先生这样的学者的努力，才使得深藏禅机的程序设计的光芒永世不灭地照耀着后代的人们。

Charlie (Chuck) Babbage

第一篇 木

大师:忍者

媒体:编年史

硬件:键盘

方向:东

软件:编辑程序

感觉:触觉

阶段:界面

动物:龙

系统:VMS

木

作为 MRVMS/IIIX 禅学院的复兴者,编程大师忍者(这是后人给他起的绰号,他的真实姓名无法核实——译注)非常有名。有一些传说,说他从一名不起眼的程序员一跃成为某家实力雄厚的程序设计机构的开发部主管。他在完成了一套先进的操作系统后,便悄然消失了。据他说,这套系统既不需要硬件也不需要软件,同样能准确地发挥功能。他现在的去处无人知晓,但听说,他时常以一位管理顾问的装束出现。

一

忍者最初来到那家开发中心工作的时候,他被分配去支持操作系统。一天,一位经理走进了忍者的工作间。

“你为什么不工作?”经理问。

“系统瘫痪了。”忍者说。

“经理皱起了眉头。”我们付给你薪水,是要你保持系统的正常运行!”他大声说。

“系统没有瘫痪。”忍者说。

二

忍者大师如是说:

“如果你的应用程序不能正确地运行，不要去责怪操作系统。”

三

忍者大师参加一次电脑展示会。

许多公司都押运出了明亮的显示器和他们所能搜罗到的所有最新、最大的硬件，以及身着泳装的金发模特儿。

忍者甚至一眼也未瞥那些展摊。相反，他只是打开一张折叠椅，静静地坐在角落里。不时地有人从他身边经过，并问他一两个问题。他思考片刻，然后用简短的话作出回答。

人们盘着腿围坐在这位大师身旁——那些收集来的宣传册散落在一边——静静地等着他开口说话。

四

忍者大师如是说：

“对于聪明的人，只要一个字；对于快马，只要轻轻一鞭；对于写得好的程序，只要单独的一个命令。”

五

忍者大师参加一次董事会。当他开始解释有关那套软件的技术特性时，那些公司行政官们，有的在座位上烦躁不安，有的盯着窗外，有的望着咖啡杯发呆。

于是，这位大师开始谈那套软件将会赚多少钱了，这时那些行政官们便纷纷竖起了耳朵，并开始房间里欢蹦着，显然，他们被这悦耳的声音吸引住了。

六

一位公司行政官来看忍者，他发现这位大师正在玩电脑游戏。“这是什么意思？”行政官询问道。

“我正在测试系统。”大师说。

行政官凑近屏幕定眼一看，果然如此。

七

一天，董事会成员集中在一起讨论业务善，他们召来忍者大师，让他作个报告。“你的季度预测是怎样的？”他们问。

“没有季度预测。”忍者回答道。

“那么你今年的预算计划呢？”他们问。

“没有今年的预算计划。”忍者回答说。

然后他们问：“你能确认你是忍者大师吗？”

“没有忍者大师。”忍者大师回答着。

董事们被搞得糊涂了，只好暂停会议各自回家去了。

八

忍者大师如是说：

“设计一个千百万程序的操作系统很容易，要改变一个人的本性却困难得多。”

九

三名来自不同的电脑公司的行政官前去向忍者学习领导艺术的真谛。

忍者问第一位行政官，“你有你们公司现在的机构设置图吗？”

第一位行政官从口袋中掏出了一张纸，“我一直带着一张在身边。”他说着便把那张纸递给了忍者。

忍者饶有兴趣地看着那张纸，“这看上去似乎已经在计算机上建立了模板。”他评论说。

“的确如此，”这位行政官骄傲地说，“我们已经把我们的机构设置图都计算机化了，以便一接到通知就能立刻选择调用。”

忍者笑着，把那张纸递了回去，“我帮不了你什么。”他说。

第二位行政官接着问忍者领导艺术的真谛。

“你有你们公司现在的机构设置图吗？”忍者问。

第二位行政官摇摇着，“我们的机构设置在这几年都未变过，我几年前就记住了。”

忍者皱起了眉头，“我帮不了你什么。”了说。

第三位行政官还是问忍者领导艺术的真谛。

“你有你们公司现在的机构设置图吗？”忍者问。

第三位行政官耸了耸肩，“我们没有机构设置图，”他说，“每个人喜欢做什么就做什么，喜欢什么时候做就什么时候做。”

忍者紧皱眉头，“我帮不了你什么。”他说。

三位行政官凑到一边耳语了一阵，转过身来对忍者说：“那么好吧，让我们看看你的机构设置图。”

忍者领着他们来到一片开阔地上，那里孤零零地长着一棵树。“这就是我的机构设置图。”忍者边说边把手搭在那表皮粗糙地树干上。

第二篇 火

大师：行者

媒体：民间故事

硬件：显示器

方向：南

软件：格式化程序

感觉：视觉

阶段：设计

动物：凤凰

系统：CP-6

火

编程大师行者收集了各种各样有关开发刺目的民间故事。最近的计算机考古研究显示，下面的这些民间故事是以历史事实为基础的。虽然一部分夸张的成分可能会无意中被添加进去，但其历史真相的核心内容依然存在。

一

当一位项目经理为他的软件项目配备人员时，他会在一天内雇佣三百名程序员。一位获得计算机科学博士学位的学者前来求职，他被给予了一个薪水很高的职位。

一天，那位项目经理被解职了。“我认为，每个人都应清楚自己的职责和履行自己的职责。”新来的项目经理在检阅他的程序员队伍时如此说道。听到这话，那位学者悄悄地溜走了。

二

两位程序员正在急诊有关用户界面的问题。

“在‘使用起来简单轻松’方面，许多意义重大的革新正在进行当中，”第一位程序员说，“不久以后，人们在使用电脑之前将不再需要阅读那些冗长乏味的用户手册了。程序都将是不言自明的。”

第二位程序员想了一会儿，然后说：“上个星期我准备劈些木柴烧火，但我的斧子又旧又钝。于是，我去五金店买了把新的。”

“这挺有趣儿，”第一位程序员说，“但这和用户界面有什么关系呢？”

“这把新斧子附带有一本长达八页的使用说明书。”他回答说。

三

一次有一位第三者问一名工程师、一名数学家、一名物理和一名程序员：“一只盒子有几个面？”

工程师首先回答。“一只盒子有四个面。”他说。

“此话怎讲？”智者问。

“四个垂直的面就是我所说的面，它们被一顶一底连接在一起。”工程师回答。

“太荒唐了，”数学家评论道，“一只盒子有六个面。”

“此话怎讲？”智者问。

“盒子是一个立方体，因此有六个面。”，数学家回答。

“不对，”物理学家说，“一只盒子有十二个面。”

“此话怎讲？”智者问。

“严格说来，有六个外部的面和六个内部的面。”物理学家回答。

智者看着一声未吭的程序员。“你的意见呢？”智者问。

“一只盒子只有两个面。”程序员说。

听了这话，工程师、数学家和物理学家大笑起来。

“此话怎讲——一只盒子只有两个面？”笑声停住后智者问道。

“这是基于个人经验，”程序员说，“那‘里面’是安置电路板的地方，而‘外面’是放显示器的地方。”

“正是如此。”智者说。

四

一位新上任的董事正在召开与程序员们的见面会。

在宴会进行当中，一位程序员背诵起了如下的演说——“我们一直都在期盼着你的到来，您的前任根本没有您这种高贵的能力。现在你来了，我们会变得真正具有生产力。”

这位新董事被奉承的喜形于色，“这演说词是你自己写的吗？”他问。

“这是我们开发中心的惯例，”那位程序员说，“无论哪位新董事到任的时候，我们都要发表那段演说，这是我所知道的唯一的一段演说词。”

五

一天，开发中心的一位程序员发现了一套可以生成迷宫图案的算法系统。他非常勤奋地对这套算法进行了修改，这样可以让她在长长的打印纸带上生成一个不间断的迷宫。

不久，他制做了一个有几百万条分岔的迷宫，四十英尺长，七英尺宽。他把打印纸带挂在了程序员办公室门口长长的过道上，不一会儿，所有编程人员都挤在了那个迷宫前，试图解决这个庞大的难题。

开发中心的主任恰巧从旁边经过，他惊愕地盯着这场景，面色阴沉。但当他走进编程大师的办公室想寻求帮助时，大师早已不在那儿了。

六

五名初学者哭着走进了大师的办公室，“呜，呜！听说我们的项目可能要被取消了。”

大师说：“所有事情照常继续，直到它们停止。”

听了这话，那些初学者便返回各自的工作中去了。

七

一天，开发中心接到消息，一位新主任将被任命来负责这里，他是位军机大臣，对电脑知之甚少。

程序员们听到这条消息都非常吃惊，纷纷停止了编程，把许多时间都浪费在揣摩那即将到来的不幸的日子。

看到这些，一位大师决定必须得做点什么。于是，他借来了一套装扮大猩猩的服装。

不久，那位大臣就职了，他把所有的经理召集到一间小会议室，随行而来的是几位总部的公司行政官，据说，他们是来使开发中心“平稳过渡”的。

突然，那位大师穿着装扮大猩猩的服装破门而入。他蹦上会议桌，把文件踢得到处都是，并对着那些行政官咆哮嗥叫，行政官们坐在那里只剩下目瞪口呆了。然后，他唰地一下离开了会议室，正如他来时一样。

听到这件事，程序员们便都回到各自的工作岗位上去了。

<作者按>

作者曾经和几个人谈过，他们都亲眼目睹过此传说中描绘的事件。作者也曾听说，一年之后，类似的挑衅行为在 IBM 的某部门中发生了。这第二起事件与第一起的不同之处在于，程序员穿了一套运动服，站在门口大声地咳嗽。

八

一组程序员正在向董事长汇报。“今年最大的成果是什么？”董事长问。

那些程序员在一起商量了一会儿，然后回答说：“我们今年解决的 bug（程序错误——译注）比去年多百分之五十。”

董事长疑惑地看着他们，很显然，他不知道 BUG 是什么意思。他小声地与身边的大臣嘀咕一阵后，转过身来对着程序员，气得满脸通红。“如此差劲的质量控制，你们应该问心有愧。明年不允许有‘BUG’！他命令道。

可以肯定，第二年那些程序员向董事长汇报的时候，有关 BUG 只字未提。

九

一位公司行政官来参观开发中心，他在长长地走廊里走着，就像一位检阅部队的将军一样。他不时地偏上来和他遇到的人谈谈话。最后，他走进了一位程序员的办公室，这位程序员正在聚精会神地为操作系统查错。

行政官环顾了一下办公室，注意到有尊猪的雕像摆在程序员的终端上。”我常常被程序员收集的那些古玩和纪念品所深深吸引，”行政官说，“在它们背后似乎

有一些非常有趣的故事。比如说吧，那里的那个雕像是什么意思？”他指着那尊雕像。

程序员从终端前抬起头来，眨眨眼睛，然后盯着那尊雕像，就好像是头一回看到它似的。“这是头猪呀！”他说。

第三篇 土

大师：隐者

媒体：说教

硬件：CPU

方向：中

软件：调试程序

感觉：嗅觉

阶段：编码

动物：牛

系统：OS/VS

土

以下语录的作者，编程大师隐者，在人们心目中的形象一直很模糊。他被 COBOL 语言标准委员会的特务所暗杀，除此之外，人们对他知之甚少。

一

我听说过：

开始时似乎比较容易的事情，结尾时往往是最困难的。这就是为什么程序员说——“开发前面的百分之九需要一半时间，而另一半时间则用来完成最后的百分之十。”

二

我听说过：

在人机界面中犯的最大错误是忽视计算机的权利。被迫与旧概念兼容的系统总是受到历史的限制。计算机不应该模拟现实——它们应该超越现实。

三

我听说过：

一家计算机公司设计了一套强大的系统，这套系统超前于它所在的时代。因为担心这套新设计可能会被复制，他们对硬件的结构设计保密，并为操作系统申请了专利。

十年以后，一位初学者向编程大师问起那个放在数据中心后面的又脏又旧の木箱子。

四

我听说过：

不要让政治原因影响技术方面的决定；也不要根据科技论文来做出政治决定。只有在无知者的头脑中，这些东西才会混到一直。

五

我听说过：

项目计划和公布的时间表，本身毫无意义。那些日期和项目进展的里程碑本质上不意味着什么。然而有一个秘密的时间表，它被所有工作于一个项目的人所理解。这个秘密的时间表从未被外界的关注所愚弄，也从未被操纵以迎合市场的方案。这个秘密的时间表总是被遵守，因为它反映了所有开发部成员之间的相互理解。当项目反映了这个现实时，程序会如期完成；当项目计划与此现实相矛盾时，程序会被延误。

六

我听说过：

有三种情况肯定会导致程序设计项目的失败。第一种情况是，主管此项目的经理对软件一无所知；第二种情况是，对程序代码负责的项目带头人对编写代码毫无兴趣；第三种情况是，编写代码的程序员是临时雇佣的，对项目缺乏忠诚。这三种情况中的任何一种都会导致项目的失败；三种情况同时出现，就必死无疑了。

七

我听说过：

许多公司行政官乐于看到把他们摆在高高在上，而让他们的下属居于其下的机构设置图。这些行政官无异于那些玩“山上的国王”游戏的孩子们。机构设置图有时候叫做树状结构，但使用那些结构图的行政官们似乎总是忘记，树叶在底下的

树是已经被连根拔起的树，活不了多久。优秀的行政官总是把自己看作是组织的询问，承受作为领导的重担，因为这才是他发挥作用的最佳途径。

这就是关于编程大师忍者的著名寓言——他没法传授领导艺术的真谛，因为那三位想当他学生的人太相信他们自己的力量了。当一个组织被出色地领导时，雇员甚至不知道他们正在被领导。当项目完成时，他们会对自己说：“看看我们通过我们自己的努力所取得的成绩吧！”

八

我听说过：

一个程序的价值不能由它的宣传册的大小，或出现在大众计算机杂志上的整页广告的数量来判断。这些噪音越响，程序越不可能有用；真正优秀的程序不需要广告，用户会口口相伟。

九

我听说过：

有一种循环，这是一种宇宙的节奏。今天一个程序流行，明天是另一个；今天修正了五百个程序中的错误，明天又将出现五百个新的。要理解生命，就要知道这种节奏的存在。要理解禅，就要活在这种节奏之外，从日常生活的牵挂之中超脱出来。只有这样，头脑才会自由。

第四篇 金

大师：大力神

媒体：公案

硬件：打印机

方向：西

软件：解释程序

感觉：听觉

阶段：排错

动物：兔

系统：MS-DOS

金

编程大师大力神试图在讲解中使用公案——禅的一种独特的教学手段。通常说来，每个公案的含意都很难辨析，并且常常有好肉食的和外在的意思。

一

一个在大学里学习了很长时间的初学者来拜访编程大师。

初学者说：“我已经记住了《计算机编程艺术》的十二卷书中所汇集的算法，我能用 LISP、PROLOG、ALGOL、MODULA2 和 ADA 语言编程。

大师礼貌的点点头。”我很少遇见像你这样博学的人，”他说，“我想听听你关于我写过的一个程序的意见。”

“我很高兴助您一臂之力。”说着，初学者骄傲地挺起了胸脯。

大师走到他的电脑前插入一张软盘。“首先我得拷贝一下。”他解释说。

他们坐在那儿，听着旋转的驱动器沙沙作响。突然电脑上显示出一条信息，“磁盘上没有空间——Abort, Retry or Ignore?”

大师敲了个“R”进行重试，他们又一次听着驱动器的声音，接着那提示又一次出现在屏幕上。大师仅仅再次摁了个“R”，重复着同样的事情。

最后初学者再也按捺不住了。“磁盘上没有空间了，”他生气地说，“他太满了！”

大师说道：“是你的脑袋太满了。”

初学者一下子恍然大悟。

二

大师如是说：“任何一个程序，无论它多么小，总存在着错误。”

初学者不相信大师的话，“如果一个程序小得只执行一个简单的功能，那么会怎样？”他问。

“这样一个程序将没有意义，”大师说，“但假设这样一个程序存在的话，操作系统最后将失效。产生一个错误。”

但初学者不满足。“如果操作系统不失效，那么会怎样？”他问。

“没有不失效的操作系统，”大师说，“但假设这样一个操作系统存在的话，硬件最后将失效，产生一个错误。”

初学者仍不满足。“如果硬件不失效，那么会怎样？”他问。

大师长叹一声。“没有不失效的硬件，”他说，“但假设这样的硬件存在的话，用户就会想让这个程序做一件不同的事，这件事也是一个错误！”

没有错误的程序是一则谬论，世间难寻。假设存在着一个没有任何错误的程序，那么这个世界将会不复存在。

三

象牙塔里的魔术师前来拜访编程大师，像往常一样，带来了他最新的发明。

编程大师从他的终端前抬起头来，看见魔术师站在门口，“这次你带来了什么呀？”大师问。

“这是我最伟大的发明，”魔术师说着，用车推着一个盒子进了屋，“这是终极桌上印刷系统，有了它，排字机、描图桌和照相机能做的事我全能做。我已经设计好了，任何熟悉传统印刷术的人能够在几个月内学会。这儿有我制作的样张，看，它像《古腾保圣经》的首页。”

大师仔细地看了看样张，露出一丝微笑。“这部装置打算用来干什么？”他问。

“我们将用它来印刷技术说明资料。”魔术师说。

<作者按>

这位魔术师经常出现在禅的民间故事中，作为对立面。这个人物是否真正存在，不得而知。然而有传说说，这座象牙塔曾被一伙狂热的信徒控制，这伙人被唆使去干软件盗版、谋杀以及其它各种各样的不道德的事情，以报复机器时代的Cray-2 超级计算机的统治。

这座象牙塔在第三个朝代末被包围并几乎毁掉，但由于政府的援助而幸免于难，并得以重建。

四

一位初学者走进大师的工作间，看见一台新电脑摆在大师的桌上。

“那台电脑是什么？”初学者问。

大师把手搭在一个小盒子上，这个小盒子通过一根电缆与那台电脑相连。“看，”大师说，“这部装置控制着我们在屏幕上看到的東西。”

初学者紧紧地盯着屏幕，但他看到的只是毫无意义的符号。

“这个屏幕模仿的是一张桌子，”大师解释道，“比如说，屏幕的这里是文件柜和垃圾桶，这里是打字机和计算器。”

“这是一项伟大的发明。”初学者敬畏地低声说道。

“并不像看上去的那样伟大。”大师说，他推着初学者的肩膀往后退了几尺，“你能看见那两张桌子吗？”大师问道。

初学者点点头。“一张在地板上，另一板在屏幕上。”他说。

“是这样的。那么，是否有一张桌子上少了什么东西？”

初学者沉思片刻。“其中一张桌子上没有电脑。”他说。

大师摇摇着。“任何一张桌子上都没有电脑。”

五

一位初学者问大师：“程序设计的真正含义是什么？”

大师回答说：“饿的时候就吃；困的时候就睡；当时机恰当时，就进行程序设计。”

六

编程大师走进初学者的工作间，听见了一阵骚乱。

“这些该死的电脑！”初学者气愤地叫嚷着，“要想让它们做点事，我得使用三个甚至四个编辑程序。有时候我被搞得糊里糊涂，结果删除了整个文件，这简直难以忍受！”

大师盯着初学者。“那么你打算怎样补救呢？”他问。

初学者想了一会儿。“我将设计一个新的编辑程序，”他说，“一个将取代所有这些程序的程序。”

突然，大师敲了一下初学者的脑袋。这一击并不重，但初学者却很惊讶。“你这是什么意思？”初学者惊问道。

“我根本不想学另一个编辑程序。”大师说。

初学者一下子就恍然大悟了。

七

一位初学者问大师，“每当我在一套新的系统上编程时，必须学会一种新的语言。为什么没有一套标准呢？”

大师转身而去。“唯一真正的标准是死亡。”他说

八

一位初学者问大师：“在 ADA 编译程序中有没有佛性？”

大师回答说：“你曾注意到在八进制、十六进制和十进制中字符 NUL 都是 000 吗？”

初学者一下子就恍然大悟了。

九

名叫狄阿古和古多的两位程序员正在向大公司的董事长汇报。

当古多站起来发言时，他对那位公司行政官说，“你天生聪明，能够理解电脑的真正含义。”“瞎扯，”狄阿古说，“你为什么表扬这个蠢才？他也许是行政官，但他对电脑一无所知。”

行政官对他们俩都给予了奖励，然后雇了一名顾问来决定他们俩谁对谁错。

第五篇 水

大师：阿幸

媒体：俳句

硬件：磁盘

方向：北

软件：编译程序

感觉：味觉

阶段：测试

动物：蛇

系统：Unix

水

尼姑阿幸最初是写俳句的。俳句即短诗，用一套预先好的章节来试图表达一种情感或一种想法。阿幸的俳句则更多地表达了一名程序设计者的那种孤独的生活：

为了调试而工作到深夜，徒劳无获的测试，看着一段程序编译时的那种简单的喜悦，等待。所有的程序设计者都有过这些体验，但他们很少提及，即使是在他们之间。

本书这最后一部分，显然是一部大作中的片段。作者不才。未能将这些短诗中的深奥的意蕴完全翻译出来，以达到和原诗类似的效果，还望读者多多包涵。

一

程序不过是梦，
生于无形无象的禅中，
我们只是那做梦的人。

二

我编程三日，
两耳不闻人声，
只有硬盘在歌唱。

三

编译程序在运行，
像一条湍急的河流，
我静静地等候。

四

为何要用 LISP?
COBOL 语言我不懂，
所以我用 LISP。

五

编程一整夜，
透过窗户，照到屏幕上，
初升的太阳。

六

硬件和软件，

我们是你们的主人和奴隶，

我们的生活是程序。

七

你能听见吗，

上程序的声音？

答案是 NUL。

八

电子邮件说，

项目已取消，

清理文档时，我哭了。

九

计算中心空空荡荡，

静静的只有风扇在嗡鸣。

我走过一排排的 CPU，

磁通量刺痛了我的皮肤。

我打开一扇门，又冷又硬，

看见指示灯在控制板上跳动。

一台没有灵魂的机器，人们说，

但它的灵魂是同事们的汗水，

在它里面有我们多年的生命，

失望，友谊，悲伤，欢乐，

算法成功时的狂喜，

和无数个徒劳无获的长夜。
我听见了叹息和欢笑的回声，
在这间黑洞洞的办公室里，
终端机闪耀着光芒，就像星星一样。

《计算机寓言 - 信息时代的启示》

作者: Geoffrey James

小赵 录入整理

声明:

本文内容来自《编程之道》一书，小赵只负责录入，所有版权归该书的出版社。
本文由原来的三册书组成：《编程之道》 《编程之禅》 《计算机时代的寓言》

引言

(一次前前往西藏拉萨的超越机器学院的旅行。)

虽然我的 C. P. Yu 博士通信交往已有多多年，但我从来没有见过他的面。去年《编程之禅》出版后不久，他给我的信停止了。几番查询也无济于事，不知道发生了什么。我别无选择，只好长途跋涉去拉萨看看我的良师兼益友是否安危无恙。

这次旅行由五次乘飞机，两次乘火车和一次骑大象组成，平安无事。不久我发现自己正沿着陡峭的石头小路往上爬，这条小路直通座落在云层之中的西藏大学。

小路的尽头是一扇厚厚的石门，门外挂着个铜钟。我双手握紧钟舌重重一击，钟声在山峦之间回响萦绕。一群栖息在附近草丛中的鸟儿，被这深沉的钟声吓得振翼拍翅，像一阵和风似的从头上飞过。

此时，一位身穿褴褛长袍的老者从门边出现了。“这个入口仅仅是为举行仪式准备的，”他说，“请随我走学生入口吧。”

我随他来到旁边的一扇小木门前，刚要抬脚进门，老者用他的小手指顶在我的胸口上，把我给拦住了。“你为何来这里，西文的野蛮鬼？”他礼貌地问。

“我是来看 Yu 博士的。”

老者扬了扬一条眉毛，然后让我进去了。

这里和我所期望的大相径庭。在我的脑海里，早已勾画出了一套具有现代中国传统风格的简单建筑。但是恰恰相反，我看到了一排排拱塔，塔身包裹着褪了色的红漆。沿着四周的墙有许多壁龛，里面供奉着古代的神像。其中一个雕塑尤其令我记忆深刻。它是位女士，从身躯里伸出许多臂膀来，每只手上都拿着一张卡片，卡片上戳有一些微型方孔。屋子主人注意到我对此很感兴趣，就在一边说：“老了，很老了。没有人知道她的名字。”

我们爬上了一条楼梯，尽头是一扇小铁门。主人用一张塑料安全卡把门开了。一排杂乱的古代计算机设备展现在眼前，奇异的磁盘驱动器和古旧的主机，被大衣挂钩和裸露的电缆缠绕在一起，计算机的机箱几乎被埋在了厚厚的灰层里，一堆发霉的 MS-DOS 使用手册散落在一边。很明显，这个房间已经很久没有人进来了。

“Yu 博士死了，”主人说：“这里是他度过他最后时刻的地方。当时在操作系统里有一个错误，他没能来得及排队这个错误就。。。。”他的声音渐渐小了。因为无法控制自己的感情而感到羞愧，他转身下楼去了，把我一个人留了下来。

正在我想跟他下楼时，听见一阵奇怪的噼噼啪啪声。这是一台原始的打印机，不知怎么回事，它竟然被满打满算了。一条慢慢的纸流从窄槽里漫了出来，落到布满灰尘的地板上。然后，随着一阵呼哧呼哧声，打印机停了。我几乎不知道自己

干了些什么，便糊里糊涂地收拾起那些纸张，匆匆离开房间。铁门在我身后吱吱呀呀地关上了。

老者正在楼底上等我，他领着我穿过迷宫式的院子，然后把我半推出了门。“下次带些多余的内存条来，”他说，“我们这里很紧缺。”接着便在我身后把门关上了。

下山的路走了一半，我才想起来看看打印机上的内容。那是一套寓言集，很明显，它是 Yu 博士最后的作品。我很快便意识到，这些寓言是他的遗言，是他对信息时代的启示孜孜不倦进行探索的记录。

第一篇 秋

虚幻与真实

1 彩虹

一间大办公室，里面满满地有许多工作间。每个工作间都坐着一位男士或女士，身着蓝色的服装。电话铃声此起彼伏。电脑屏幕发出的琥珀色光线映在无色的天花板上，非常刺眼。

窗外，暴风雨来临了，雨点砸在地上，狂风猛烈地摇着树梢，雷声运动着这幢黑色砖墙建筑物。

男士和女士们根本没看见这场暴雨的到来，也没听见狂风的呼啸。

突然，灯光暗了下来，屏幕上变成了一片空白，电话铃声哑了。人们咒骂着，像一只只小虫从那些方形的茧包中钻了出来。

人们聚集到一起，谈论着，笑声充满了整个走廊。

一位男士，站在窗外户边看着这场暴雨过去，一道彩虹金光闪闪地挂在东方的天空上。

2 机器人

一位科学家用他自己的形象制造了一个机器人。科学家坐在自己的造物前，看着它脸上奇怪而熟悉的笑容和丰富的表情，恼了。他想测测这个机器人的智商，便问机器人：“这个宇宙始于何时，又将于何时轮回？”

机器人回答说：“这个宇宙在我出生的时候被创造，在我死亡时它也将结束。当我行走时，我并不移动——相反，是世界在我的脚下移动。当我闭上眼睛时，除了虚空以外不存在任何事物，但当我两次睁开眼睛时，世界便一眨眼又回到了现实中来。没有什么东西是真实的，没有什么东西是永恒的。当我离开一个地方时，这个地方就像烟雾一样散去；只有当我回来时，它才又重新出现。除了我本身以外，没有任何东西存在，我就是虚无的上帝。”

科学家得意地点点头，“我明白了，你毕竟只是台计算机而已。”

3 专家系统

一位计算机科学家和一位年轻的女士结了婚。科学家知道他早她去世，于是开始营建一套专家系统。这是一套能录制他所知道任何事情的程序。当系统完成时，他对妻子说：“从现在起你将永远不会孤单。”

不久以后他死了。他的妻子从葬礼上回来后，把那套专家系统调到了电脑屏幕上。她那死去丈夫的面容出现了。她对着屏幕讲话，屏幕便立刻作出了回答。

她流着泪将那套程序从电脑中删除了。当有人问她为什么时，她说：“我太爱我的丈夫了，以致于不能用那种方式来想起他。”

4 电脑色情狂

以前有一个男子酷爱色情画。他买了一台昂贵的电脑，这样他就能观看高分辨率的女人裸体照片。他亲热地给他的电脑取了个名字叫“安琪儿”，并花费大量的时间来观看他所收集的照片。

不久，他便对现实中的女人失去了兴趣。当有人问他为什么时，他回答说：“我的电脑是我创造的世界，它顺从于我的每一个念头。当我坐在屏幕前，我就像一个上帝，为什么我要浪费时间在现实中那些不合意的或纠缠人的女人身上呢？”

一位程序员听到这话，抱怨道：“为了这样的一种基本目的而使用电脑是完全错误的，你应该活在现实的世界里。”

那个色情狂便问程序员：“顺便打听一下，你用你的电脑干些什么？”

程序员骄傲地挺起了胸脯，“我编写程序来模拟经济趋势。”他说。

5 新闻发布会

一名市场营销员和一名程序员都在一次新闻发布会上发言，将一项有关新的电脑技术的消息公布于众。

市场营销员说：“这项技术比原子弹、晶体管和电话三项发明加起来对世界文明的影响都要大。”

程序员则是说：“这项技术在有限的领域内，在有限的程度上，解决了一些技术性的问题。”

猜猜看，记者们将会引用谁的发言。

6 神符

从前有个巫师，他非常强大并且功力深厚。一天晚上，一轮昏暗的月亮出没在天空，一尾彗星像不祥的眼睛挂在西边。此时，巫师制作了一张力量的神符。在那神符上面有许多圆圈、方块、线条、箭头和词——有魔力的词，如建筑、信息、自动化、集成、分销。

然后，巫师把他那飘逸的乡满银色的星星的长袍扔在一边，穿上了羊毛三件套，朝硅谷奔去。

他走进一位军机大臣的办公室，说：“看啊，用这张神符，我们将捕获人们的思想。不计其数的黄金将飞进我们的保险箱，不可限量的电脑资源将归入我们的项目。”

军机大臣拉开他的抽屉，翻出了一大堆纸张，每一张上面都有不同的神符，“我的神符已经够多了，”他说，“我所需要的是一套好的，稳健的和有用的程序。”

巫师只好回去了，他气得直咬牙，咒骂这次倒霉的运气。

7 垃圾和福音

一名程序员参加了一次业务讨论会。

第一位发言的人把一张胶片放在投影幻灯机上，许多组数据便在屏幕上显现了出来。“这张表格向我们显示，”他说，“我们将会赚很多钱。”

程序员问：“当你准备这张表格时，你是做何种设想？”

那位发言人挠挠脑袋，“我是在电脑上制作这张幻灯片的。”

程序员咬着牙，“你不曾听说过‘垃圾进，垃圾出’（Garbage In, Garbage Out）这句话吗？”

那位发言人仰头大笑：“你引用错了，正确的说法是‘垃圾进，福音出’（Garbage In, Gospel Out）！”

8 人工双脚

一位著名的科学家正在预测电脑的未来：“在人工智能方面的发展，将导致电脑在各个方面都与人平等。电脑将会被编程以便能够作决定，并提供定于想像力的解决方案，甚至感受人的情感。”

那天晚上，科学家梦见他的双脚变成了两只轮子。

9 考试

一位编程初学者问编程大师：“幻觉和真实之间的区别是什么？”

大师考虑了一会儿，然后让那位初学者在屏幕上敲入单词“THUNK”（think，计算机中的形式实在转换程序——译注）。“现在大声朗读这个单词。”大师命令道。

初学者读道：“TUNK”。

就在这里，大师重重地敲了一下初学者的头。

初学者揉着耳朵问道：“你这是不是意味着，疼痛是真实的，而屏幕上的那个单词却不是呢？”

大师叹了口气，“孺子不可教也。”，他说。

10 示意图

编程大师叫他的一位初学者画了张计算机系统图。真实这位初学者画了张软件图，但图上没有标示硬件；然后他画了张硬件图，但图上又没有标示用户；接着他画了张用户图，但又没有标示功能；于是他又画了张功能图，但却没有标示软件。最后这位初学者沮丧地放弃了，并对大师说：“这个系统太复杂了，根本无法画在一张图上，它不可能在同一时刻被全部理解。”

大师听到这里，笑了一会儿，随后开始哭了起来。初学者问他为什么，他回答说：“你认为计算机系统很复杂，但与一只的眼睛相比，它还不算复杂。繁衍生命的地球的系统要远远复杂得多。而在数十亿个太空世界中，地球本身仅仅是一颗行星而已。”

“初学者说：“我还是不明白，你为什么笑了又哭呢？”

“这还不明显吗？因为资本主义和共产主义之间的区别，这就要引发战争了。想想看——那些经济理论系统顶多是浩瀚宇宙中无足轻重的一方面的不完整的图表。然而，有些人却因为在某种抽象的教条上产生了就要毁掉这个地球。这不可笑吗？这不可悲吗？”

11 树叶

一位公司行政官花了一上午时间听取其下属的汇报，在每个汇报结束时，汇报人都交给他一叠写满实际情况和数据的文件。

到了中午，行政官把那堆文件塞进了他的公文包，向大家匆匆致谢后便离开了大楼。时值深秋，片片树叶被大风从那半秃的树枝上扯摘下来。

突然，他的公文包的锁扣断开了，那些文件被风摇着，翻腾着，同那些黄色的烂树叶混在一起，像一群受惊的鸟飞散开来。

第二篇 冬

战争与死亡

1 盖娅

盖娅对她最聪明的孩子曼说：“你将成为我的才能的管家。”曼投入盖娅的怀中，对她非常尊敬。

但是，一个邪恶的念头闯进了曼的脑海：“只有我的身体是属于盖娅的，我的灵魂属于来世，并且将从此轮回。”于是曼不再崇敬盖娅。

然后，又一个邪恶的念头闯进了曼的脑海：“因为我比盖娅强大，所以我应让她成为我的奴隶。”

于是曼用庞大的刑具来折磨盖娅，他弄破她的皮肤露出了骨头，把浓烟灌进她的鼻腔里，泼脏水到她的胸上。

盖娅对曼说：“你冒犯我，后果自负。”但曼听不进她的话，那些话对他来说是耳边风或海里的潮汐。

于是，盖娅抬起手，将灾患一一吞食生命的灾患降临在了曼的身上。然后，盖娅流着泪召集她的那些不听话的孩子们回到了她的发源地，回到了那个寒冷的却又舒心的地球上。

2 战争机器

一位将军对一位考古学家说：“看啊，这台已经造好的机器，它能在片刻之间便招来死亡，它的一小块铁片要比古代的珍宝昂贵得多。这不是奇迹吗？这台毁灭来者机器？”

考古学家拍着那台战争机器冰冷的金属外壳，非常难过，就像一个人拍着即将离别的朋友的脸庞。然后他说了这样一番话：“一天，我走过一片空地，我发现了一把又旧又钝的剑，我拾起它，但它在我手中腐烂了，不值一文。”

3 两位老人

两位老人坐在公园的长椅上，谈起了过去的时光。

“当我年青的时候，”第一位老人说，“我是海军士兵，我在每个港口都有个女孩。当我年纪大点时，我是名推销员，我在每个城镇都有个女孩。当进入中年时，我又花钱买来许多漂亮妓女服务。现在我满腹哀，因为我不久就要死了，再也尝不到女人肉体的快感了。”

“我的生活却不同，”第二位说，“当我和我的妻子刚刚结婚时，我们一个晚上做两次爱，对于刚刚开始的爱激情我们不知所措。随着我们年纪的增长，我们做爱频繁了，但每一次爱抚都充满了缠绵的快感。当进入中年时，我们很少做爱，但我们的灵魂就好像融为一体。现在，我充满了喜悦，因为我将在死亡那无穷无尽的永世，永远地爱她。”

4 影子武士

一位初学者对编程大师说：“对于我们所编的一些软件，我感到越来越不安。如果我们的程序被用在战争的武器上，那么，我们不是要为因此而导致的死亡负责吗？”

大师回答说：“在未来，战争将要依靠计算机，而不是钢铁和肉体。那些迷失于茫茫数据之中的将军和政客们，将忙于模拟和幻觉而不能自拔。那里，世界将得到拯救。”

5 医学研究者

有一位科学家常用动物做实验。他全神贯注于他的研究，笼子里动物惊恐的叫声，他充耳不闻；活生生的大脑被剖开后用夹子夹着，他视而不见；强腐蚀性的药物胶布下面散发出的腐烂的肉体的异味，他似乎也没闻到；当他的实验品死了时，他也感觉不到解脱。

一天，这位科学家得知自己患了癌症。“我多么高兴呀，”他想，“诸如我之类的科学家经过长期劳作，现在终于找到了治疗这类疾病的办法。”

六个月后，这位科学家躺到了医院的病床上。受化学疗法的影响，他的头已经秃了，身上的褥疮浸透了纱布绑带。一根根管子插进了他的血管，一副氧气罩往他的肺里充着气。

死神就坐在床边，没被人看见。“我认为，我应该让他活得更长些。”死神说。

6 舰载系统

有一艘军舰，装备了一套尖端的计算机系统。一天，这艘军舰击落了一架民航客机。海军的一位代表说：“我们的系统极为先进，不会出任何差错。尽管这套系统从未被调试过，它也没有出差错——是一位年青军官出的错。”

一位程序员听到这话，简直不能相信自己的耳朵。“只有凶手才可能会这样说，‘我是清白的，那不是我——那是我的手钉的。’一套计算机系统由三个部分组成：硬件、软件和人。如果人犯了错误，系统早就已经失效了。”

7 宇宙飞船

十三名科学家聚集在一起，互相谈论着：“我们已经给地球千万了无法修补的污染，让我们再寻找另一个世界来居住吧。”

于是他们建造了一艘宇宙飞船，飞船载着他们穿过空荡荡的太空。这个航空器呈圆形，里面所有的东西也是圆的。废物经过处理后变成了食物，即使空气中的水分也被收集起来再利用。靠这样的方法，他们试图模拟地球的生态系统。

那些科学家在太空里航行已经有很多个月了。一天，废物处理器破裂了，剧毒的沉淀物涌了出来。疾病开始在那些科学家中间滋生，他们一个接一个地都死去了。

8 博物馆

有位富人一次把大笔的钱花在制造低温箱上，他想让他的身体能在他死的时候被冷冻住，并能在未来的某个时候复活。

两个星期以后，他参观了一座博物馆。在那儿，蜡封的玻璃棺里，他看到了法老们干巴巴的、洒了香水的尸体。这些尸体被展示出来，使人大开眼界。

9 圣诞节的故事

一名程序员编写了一段程序，这段程序绘出了一棵圣诞树，并能同时演奏“铃儿叮当”的歌曲。

碰巧，圣诞老人从旁边经过。程序员全神贯注地盯着屏幕，并没有觉察到这位精灵身穿的飘逸的绿色长袍和几乎拖到地上的长胡须。

这位老精怪从程序员的肩膀上望过去，看到那个屏幕，心里想：“在我年青的时候，情形和现在截然不同。我还记得，那时候猎人们都是光着脚在雪地里奔跑，用尖锐的石器宰杀野鹿。猎人们把野鹿的内脏吊在青树上，目的是为了野鹿还生，然后他们点燃一根大木头，在火上烤着野味，唱着圣歌。那段过去的时光简直真是太美好了。”

10 去世的顾问

两位计算机顾问正在谈论着一位刚刚去世的同事。

“我简直不能相信，”第一位顾问说：“他以前看去很有活力。他知道五士多种文本编辑程序和一百多种编程语言。他有一间屋子，里面堆满了各种各样口算生产厂家的个人电脑。他属于所有网络，每一天收到一百多封电子邮件，并认真答复每一封邮件。他秘书科被计算机革命通上了电源，拧开了开关。”

“对于他的过世，我也感到很吃惊。”第二位顾问说，“顺便说一下，他怎么死的？”

第一位顾问摇了摇头，“他得冠心病。”

11 玻璃城

在玻璃城里，一位公司行政官正站在一座塔的顶上，他说：“看看我们所建造的一切吧！我的城市，是时代的奇迹，如今，世界上的事务要在这里处理。晚上，街灯点亮了天空。是的，我是众生之首。”

死神就站在这人身边，没有被发现。死神低声道：“在非洲有许多白蚁堆，远比这座城市复杂多，而它们处理的事务与你们处理的事务相比，丝毫不显得渺小。高高在上，永恒的星星点亮了无边无际的天空。更何况，你那城市的灯泡还不如一只小虫的屁股发出的亮光呢！”

然后，死神继续赶他的路因为他和墨西哥的一场地震还有个约会。

第三篇 春

真爱与新生

1 两名学生

曾经有一名学生，他在课余时间穿上古时候的盔甲，随后他如今那些和他有类似想法的人在一起，模仿古代社会里的贵族和太太们。

另一名学生是计算机科学专业的，他问一名学生为什么要穿着奇装异服走来走去。后者回答说：“二十世纪是多么的枯燥无味，你花那么多时间呆在冰冷的机器中间，而我在寻觅过去时代的美丽和浪漫。”

计算机专业的学生淡淡一笑，领着第一名学生进了数据中心。“这里有你所蔑视的机器，它们不比你的长凳更雪亮吗？它们在全副武装的铁箱里不是很安全吗？三百年之后，孩子们将会装扮成以前的计算机科学家。”

2 一对猎人

一名程序员一次发明了一套经济电脑系统。当这套系统完成了并被安装到客户指定的地点后，他前去拜访用户，期望能够得到他们的感激。

当他走进大厅，发现一位漂亮的女士正坐在一台终端机前。“你觉得这套新的系统怎么样？”他问。

女士叹了口气，凄凄婉婉，“在我们安装这台电脑之前，”她说，“我必须和许多人说话，并执行不同的任务，可现在，我除了是一台终端机的附属物以外，什么也不是了。我的工作成绩是根据我每天进行多少次屏幕演示而定的。有时候，我的背简直酸得直不起来。”

程序员沉默片刻，然后说：“我是这个电脑系统的发明者。对于这套系统的创建，我很兴奋。我必须运用我的一点一滴的技巧和知识来使这套系统易于使用。得知你对它很不，我感觉非常难过。”

女士触摸着他的手。“不要责怪自己，”她说，“你活在另一个世界里，你怎么会知道我们的情感和心愿呢？”

真可谓水到渠成。那天晚上，他们两个人做了爱。深夜，那名程序员醒了一会儿，他躺在床上，眼睛盯着天花板，他那熟睡的情人的头在他的胸膛上轻轻摇晃起伏。

3 公司行政官的幻境

有一位公司行政官，他有幢很大的房子，一辆非常昂贵的汽车，一台速度很快的电脑，一位富有的妻子和一个可爱的懒妇。他从一个城市飞到另一个城市出差，住最好的宾馆。他白天整天喝咖啡，到了晚上就喝威士忌。

一天，他坐在飞机上，飞机正做着令人困乏的远程的航行，他的眼睛瞟向了窗外。零散的云朵在田野里投下斑驳的影子。他朝下望去时，似乎觉得他所过的生活像下面的小房子一样遥远，像公路网漫无目标地通向四方。

幻境过去了。当他着陆并去酒吧要了几杯烈酒后，甚至幻境的记忆也消失了。

4 航行者

一名程序员曾创建了一套庞大的数据库，它包含了世界上有关文学、史实、人物和数据的所有资料。然后他又建立了一套先进的索引系统将那些知识联系起来，这样，他便能随心所欲地在数据库里航行。他坐在电脑前享用着他的劳动成果，即满足又愉悦。

三分钟过后，程序员开始头痛；三小时过后，程序员感到很不舒服；三天过后，程序员毁掉了他的数据库。当问他为什么要这样做时，他回答说：“那套系统把这个世界置于我的指尖，我能到任何地方，观看任何事情。因为我不再被外界的条件所限制，所以，我没有理由不知道任何存在着的应该被知道的事物。我吃不下，睡不着。我所能做的就是数据库里游荡。现在我可以休息了。”

5 正在工作的父亲

一个小孩在学校里学到了有关计算机图形学方面的知识。当他回到家里，他发现他的父亲正在一台手提电脑前敲动键盘。”你在干什么呀？”小孩问。

父亲不想被打扰，“我正在工作！”他大声吼着。

小孩垂头丧气地盯着地板，“我以前并不知道你能用电脑工作。”他说

6 智力游戏

一家电子游戏公司雇了一名顾问设计世界上最难的图形智力游戏。

这位顾问冥思苦想了很多天，最后他胳膊下夹着一只盒子回来了。他一边往里走，盒子里一边哗哗作响。

“你已经发明出了世界上最难的图形智力游戏吗？”董事长问。

“是的。”顾问打开盒子，往董事长桌子上一倒，倒出了上千张一模一样的黑色方块绝片。

董事长疑惑地看着这些方块，“我不明白。”

顾问说：“有再种方法可以使一个图形智力游戏玩起来很难。第一种是要有类似的形状，第二种是要有类似的颜色。在这个智力游戏中，每张绝片都有相同的开关和相同的颜色。难道还有比这更难的了么？”

“但是，如果所有的方块都一样，那么它们怎样排列都可以，即使一个小孩也能够把它们排列在一起。”

顾问一本正经地点点头，“在生活中，就是这样——最困难的问题往往最容易解决。”

7 电子邮件系统

一位公司行政官思量着使他的员工工作起来更有效率，于是安装了一套电子邮件系统。

两个星期以后，行政官停止了这套系统的运行。当有人问他为什么时，他说：“只要两个员工之间一产生小小的分歧，第一个员工就发一封邮件。第二个员工读后便会写一封回信，并发一份拷贝给他的经理，然后第一个员工又写另一封邮件，并把拷贝发到大楼里飞来散去，每个人都没法完成工作。”

当我的员工成熟了，能就会新技术了，我将让他们重新使用这套电子邮件系统。”

8 鸭子

一位经理常常对她的小女儿说：“妈咪必须拼命地工作，所以，当妈咪晚上回到家的时候，你要好好听话。”

一天，保姆生病了，这位经理不得不把她的女儿办公室去。她的办公室在一个工业园里，园区中央是一个很大的鸭塘。小追逐着给小鸭子喂食，红通能宾脸上充满了欢笑。

当那位经理又一次对她的女儿说：“妈咪工作很忙，你要好好听话”时，小双手叉着腰，小眉毛紧锁着，倔强地说：“但妈咪。。。你是和小鸭子一起工作。”

两个星期以后，那位经理正坐在会议室里开会，突然忍不住笑了起来。

9 老太太

一名程序员在路上遇到了一位老太太。这位老太太正在编着藤椅，她那双饱经风霜的手飞快地缠绕着，从某种程度上说，藤条的粗糙更加有助于而不是妨碍她干着手中的活儿。

“那是一张漂亮的椅子，老太太，”他说：“藤条的展示出交叉对立的矢量的相交，极富意味地表达了一个多元二次方程式。”

老太太甚至没抬眼看他，“手里有活儿，心里有上帝。”她说。

10 各有所思

一名市场推销员正在谈论电脑业的轶事。“一次，我遇到了编程大师。”他说。

“真的吗？”有人问，“他长得什么样子？”

推销员沮丧地摇摇头，“我记不太清了。当我告诉他我们能赚多少钱时，他没吭声；当我告诉他让他编写什么程序时，他紧盯着他的终端；当我告诉他计算机工业接下来将发生什么变化时，他一句话也没说就走开了。”

此时，推销员的眼睛向下瞧去，在他脚边有一只猫正盯着他，它的头好奇地歪歪地翘着。

11 立碑

曾经有一名程序员在一个遥远的开发中心工作，中心座落在一个荒凉的岛上。一天晚上，他睡不着觉，便起来到户外走走。

算法和公式像不散的幽灵在他的脑海里回旋，不断提醒他那些尚未完成的事情，他感觉就像肩膀上正找着巨大的重担，因为公司的前途依赖于他的成功。

这是一个晴朗的夜晚，一轮凸月把道路涂成灰白色。他来到了一处荒地，发现了许多立碑，这些立碑排成一个圆圈。这是已被人们遗忘的民间传说中所提到的一处旧址程序员想起了他曾经在书上读到过的什么事情——这些立碑曾经是天文计算器，用来预测天体的运行周期。他抬眼望去，看见一尾声流星就像一根银针划过天空。

无比的兴奋在他的心里涌动，他开始大笑起来。他的笑声在空旷的田野里回响，吵醒了在那些白色石头废墟下酣睡的古人们的好梦。

第四篇 夏

启示

1 蜘蛛网

一位军机大臣长途跋涉前来拜访编程大师。军机大臣原以为这位大师既有钱又有势，但他失望了。了看见大师身穿 T 恤衫和蓝色的牛仔裤，正端坐在一间小小的工作室里。

身穿三件套的军机大臣又臂交叉抱在胸前，向下注视着大师。“你因头脑精明而著称，”军机大臣说，“那么，我问你，计算机的秘密是什么？”

大师抬起头看着军机大臣，一句话没说。随后他领着军机大臣来到花园，那里有一只蜘蛛已经在两根之间吐织出了一张网。露珠挂在网上，阳光被反向成千万条细细的彩虹。

大师低声说：“仔细看着。”然后他拾起一根细枝，轻轻碰了一下蜘蛛网，那张网便像个生灵一样颤动着。“这就是计算机的秘密。”他说。

军机大臣认为大师在嘲弄他，便捡起一块石头朝那张网扔去。石头穿网而过，那张网只留下了破碎褴褛的丝缕挂在潮湿的叶缘上。他走了，与来的时候相比，他一点也没有变聪明。

2 两位艺术家

两位艺术家互相谈论他们各自的艺术品。

“我自己混制颜料，”第一位说，“我从地里挖出矿石来，把它们磨成粉，并用我的唾沫把石粉混合搅拌，然后把存放在一个泥坛子里。我用自己的斧子砍下一块树皮，我用这块树皮来作画。当我完成一幅艺术作品时，我知道那完全是属于我的。”

“我自己设计图形程序，”第二位说，“我从满是灰尘的杂志堆中打捞出程序和算法。我在自己的键盘上输入数学公式，用这些数学公式绘出我的曲线。当我完成一幅艺术作品时，我知道那完全是属于我的。”

3 一则古老的寓言

董事长命令他的程序员制造一台功能强大的巨型计算机。

这名程序员工作了许多个月。终于，他领着董事长进了一间堆满铁盒子的房间，这些盒子上镶嵌着闪闪的指示灯。“这就是我按照您的旨意建造的机器，”他说，“它能回答任何问题。”

董事长被吸引住了，他决定问这台计算机一个自从时间开始以来就困扰着人类的问题：“上帝存在吗？”

程序员把这个问题敲入了计算机。

灯光闪烁着，房间涌起了一股股电热。最后，一个深沉的声音从一个铁盒子里面发出来，“现在他存在！”

听到这个，董事长惶恐地溜走了。

程序员淡淡地一笑，心想：“那个傻瓜以为计算机说的是它自己呢！”

4 两个计算机使用者

在一间办公室里有两个计算机使用者。第一个使用者说：“我一刻也不能忍受这台机器了。当我输入命令的时候，它却不执行。真是，它按一种方式运行，但然后又按另一种方式运行。我想它肯定是恨我，因为当我迫切需要它的时候，它似乎总是出错。”

第二个使用者说：“我在我的计算机前面，就像一位端坐在宝座上的国王。我签发了一条命令，便静静地等着，看看发生了什么。当我的愿望满足了，我很高兴；

我的愿望受到了阻挠，我就试试另一条不同的命令。我把我的计算机看作是一个奴隶，我不期望它像我一样聪明。”

5 程序员的梦

有一名程序员从来不在晚上编写程序。有人问他为什么，他回答说：“一天晚上，我在我的终端前睡着了。我感觉到我的意识漂出了我的身体，顺差电缆流进了数据中心。我感到自己进入了主机，看见那些程序就像疾风一样飞过内存，我静静地惊诧于这完美的对称。我发现了一个武器，这是一条与外界相连的通道，穿过那些将信息发往世界各地的不计其数的电脑，我达到了电话系统里。我的意识变得更加平静。我一会儿无处不在，一会儿踪迹全无，不着边际。千万年一瞬即过。突然间我醒了，我的头压在键盘上，整个屏幕满是不断重复的字母‘A’。这就是我为什么不在晚上编写程序的原因。”

6 两位公司行政官

一个公司的董事长让他的两位行政官谈谈他们各自的工作观。

第一位行政官似乎认为这个问题没有必要考虑，很快就说了起来：“我的身心完全投入到我的工作当中，我整日整夜地工作，我希望我的员工也能有同样的敬业精神，我根本没有退休的打算。”

第二位行政官考虑了一会儿，然后谨慎地说：“我的每一种行为，每一句言语都经过了自身的中央处理器的处理。有时候，当我遇到难题时，我就趴在桌子上睡一会儿，醒来时便有了答案。我总是使我的组织异性到不再需要我的指引的时候，我将悄然引退。”

7 程序员的妻子

有一名程序员，他发明了很多东西，但那些发明的理念总是被别人偷窃。程序员的妻子满肚子怨恨，她问：“你为什么从不为你的那些思想去争取荣誉呢？”

程序员眨眨眼睛，就好像刚刚从梦中醒来。他说：“当我编程的时候，我处于一系列的连锁关系之中。我的想法变得具体起来，我的理念把虚幻转变成现实，存在的结构就在我眼前重新组合，我变成了宇宙创造力的载体，我腾空而起就像乘着龙的翅膀。我为什么要在乎是否有人知道我的名字呢？”

8 固执的经理

一名初学编程者告诉大师，他计划就会一位特别麻烦的经理。“在我的电脑的协助之下，我手边已经有一上亿条事实，我将写份详细的报告，那位经理将会惊奇于我的数据，并改变他的想法。”

大师评论道：“世界上所有的事实都不可能改变一个固执的人的想法。他一旦抓住了一个适合于他的本性的观念，就绝不会传诵这个观念，即使在他的脖子了架一把刀。对于这样一个人而言，上亿条事实他也会无动于衷。”

初学者挠挠前额，“既然事实如此无能为力，那么我怎样才能改变他的想法呢？”

大师说：“你必须忘记事实，你必须忘记数据。给他一个观念，这个观念要能在他的本性和世界之间架起一座桥梁，只有那里，了都会改变他的想法。”

9 两位经理

有两位经理主管同一个项目。一天，他们都参加一次计划会议。

一天的会议结束了。第一位经理萎靡不振地走到了家门口，用肩膀摔开了家门。他对妻子抱怨说：“我简直不能相信人们是多么的愚蠢。今天我参加了一个没完没了的会议，他们讨论那些芝麻点儿大的事，就像讨论生死之事似的。我当时所能做的，就是控制自己不尖叫出来。如果要我在那里再多工作一天的话，我真的要发疯了。”他的妻子被他的话吓住了。那天晚上，他一直都醒着，孤零零的。

当第二位经理回到家里，他对他的妻子说：“今天非常有成效。我参加了一个会议，会上讨论了许多小细节，但那些只是事情的表面现象。随着我们进一步的互相交谈，我发现就像一朵芬芳的鲜花在我们面前绽放。真的，我们是思想的园丁。”然后他吻了他的妻子，在余下的时间里他已忘记了白天的工作。

10 两位教师

在一条街道的人行道上，一位身着黑色套装的老头正在大声嚷嚷着引自一本圣书中的仙落。他的脸胀得通红，眼里充满着可怕的愤怒。几位身着真丝时装的年轻人从旁边经过，他们似乎什么也没听见，好像只是找到了笑料而已。

远处，一座山俯视着这座城市。在山上，一位爷爷和他的孙子坐在一起，他们静静地看着太阳渐渐西沉。正当夕阳摸着了平滑连贯的地平线时，爷爷俯过身来，凑到孩子的耳边，轻轻地告诉了他一个秘密。

11 圣人们

历代圣人们从天堂下到人间，寻找一位可靠的继承人。他们立刻便陷入了要寻找的人的类型的争论中。

释迦牟尼盘着腿坐在树下，他说：“当然，我们必须找一位僧人，只有僧人才能理解宇宙那稍纵即逝的本质。”

孔夫子捋了捋他的稀疏的胡子，“僧人只关心他们自己的发展。我们应该找一位政治家，他能制定法律，从而给社会带来秩序。”

摩西双臂交叉在结实的胸前，“政治家总是被权力的爱好所左右。我们最好找一位律师，他可以对法律进行解释。”

耶稣深深地叹了口气，“律师经常是冷酷无情的。我们应该找一位传道士，他能教诲大众。”

穆罕默德的眼里闪着火光，“吵，总是吵！我们需要一位武士，他将捍卫人类的神圣。”

老子站在一边，没有加入这场争论。其他的那些圣人都要求老子做个评判。“我们应该寻找什么样的人？”他们问。

老子一声未吭，只是带领他们沿着一条崎岖的小路行走。

小路变成了大路，接着大路又变成了高速公路，他们来到了一座城市。耀眼的灯泡和花花绿绿的广告牌遮住了地平线，电线像黑色的蜘蛛网一样在头上穿越交叉，汽车就像发疯的铁壳昆虫一样飞驰而过。当飞机尖锐的轰鸣空透那褐色的有毒的云层时，众圣人都用手捂住了耳朵。

最后，他们来到一幢大楼前。在这幢大楼里，他们看见一个人正坐在电脑屏幕前。

那些圣人用一种不信任的眼光看着老子，“一名程序员？你发疯了吗？”他们问。

老子笑了。“请看仔细些。除了他眼前的屏幕，这个人什么也没看见。被周围的电脑簇拥着，他只知道他所创造的现实，但又意识到那是虚幻。他已经超越了技术。也超越了机器。”

关于《编程之道》及其作者

当你进入 Internet, 在 Yahoo 或 Infoseek 等搜索引擎中敲入“The Tao of Programming” (本书的英文名) 之后, 你将会惊奇地发现, 这本书的出现频率是如此之高, 有 500 多个站点贴有该书的内容! 本书的流程序, 由此可见一斑。

更有意思的是, 本书中的很多格言妙语, 在美国被印在了 T 恤衫、咖啡杯等商品上, 成为这些商品促销的手段之一。可见, 这本书当时在有为风靡, 与其说它导致了一段流行时尚, 不如说它形成了一个文化热点。

迄今为止已重印了 12 次的《编程之道》, 自问世以来, 受到来自各方面的好评。

本书一出息, 《纽约时报》和《科学家》杂志便长远规划了本书的内容; 美国的许多全国性报刊, 包括《迈阿密先驱报》, 和众多广播电台, 如“美国公众放手电台”, 均调试评价了本书; 本书是美国 Prentice Hall Book Club 的特别推荐读物, 并被《电脑购买者》杂志评为最受欢迎的十本书之一。

本书的作者杰弗雷·詹姆斯(Geoffrey James)毕业于加利福尼亚大学, 1977 年开始在一家大的电脑公司里任软件工程师。他因设计和运用了一种先进的电脑辅助排版系统而荣获美国的“卓越技术奖章”, 他还为一些主要电脑供应商揣摩全球计划。

杰弗曾执教于加利福尼亚大学、华盛顿大学和波士顿大学, 教授技术与管理方面的课程, 并多次应邀在国际会议上发表演讲。他已经撰写了七本书, 最近的一本新书《电子精英的经营智慧》在初版后的三周内便销售一空。

步入中年的杰弗雷现在是 Insitute for Business Wisdom 的执行董事, 该学会倡导“应用文化动力学”的理念, 即把最好的管理技术结合进最成功的高科技企业, 并将此管理技术交流和推广到其它行业的企业。该学会的网址为 <http://www.businesswisdom.com>。若想对作者有更多的了解, 除了些站点外, 您还可以访问 <http://www.upside.com> 或 <http://www.datamation.com> 来检索他最新的文章, 这些文章以论述高科技产业的发展趋势, 及其企业的管理和产品营销为主。

杰弗雷对中国的文化和历史特别感兴趣, 尤其是太极拳。他每天早晨起来都要练太极拳, 已经有 20 多年了。据作者本人说, 他当初写这本《编程之道》的时候, 其灵感的产生和构思的形成, 得益于每天早晨太极拳练习后的那种玄妙的感觉。