# 计算机科学一百年 - 文章 - 伯乐在线

计算机科学和数学的论文读起来向来很具有挑战性，但读完后收获也不小。我也并不总能够完全理解论文中的每一个批注，有时都无法完全明白作者的结论，但阅读它们仍然大大地开阔了我的眼界。

我是比较晚才开始意识到阅读学术论文的重要性的。当还是学生的时候，我记得我只读过两篇论文，其中一篇下面我会提到。作为程序员，我并没有很强的计算机理论背景。学术论文对我来说非常生涩和遥远。因而，我花了很长时间才意识到没有阅读这些论文对我的损失有多大。

我有些同事知道了我最近对学术论文有所研究后，都在问我建议他们从哪里开始。看了Michaels Feathers和Fogus做的一张类似的清单后，我也编辑了一份自认为代表了过去100年计算机科学发展历程的清单。在编辑的时候，我采用了如下的选择标准：

- 这篇论文必须改变了世界
- 这篇论文必须颠覆了我当时的既有观点
- 每十年只能有一篇入选

这样的选择标准一定是非常苛刻也非常主观的。如果你认为我漏掉了什么，也许你也可以编辑一份你的论文清单。

直觉主义的建立早于“计算理论”，但前者对后者的建立至关重要。Luitzen Egbertus Jan Brouwer的早期构想其实是对证明法的一个批判，但是Arend Heyting在后来的工作中把直觉主义结合应用到了数理逻辑中。

我选择Brouwer的论文是因为它抛弃排中律强调价值的建设。Brouwer并不认为一个命题如果不为假，则必为真。他认为，命题必须要被严谨地正面证明为真才对。

作为一名程序员，计算一个值和仅仅表示这个值的存在两者之间之间的区别是非常紧密相关的。更重要的是，起源于Brouwer想法的构造数学在随后的类型论发展中起了至关重要的作用。

Alan Turing之所以在编程社区之外被众所周知，是因为他在第二次世界大战中破解了密码系统，和他所遭到的迫害而且最后自杀身亡。这篇论文诞生于战争之前，当时他还是个年轻的博士生。

我发现自己很难用直觉来理解图灵所描述的一个现象——不可计算数。我对于认为计算本身有硬性限制，这种说法既让人费解，又充满魅力。

在著名的判定性问题上，尽管图灵机和Alonzo Church的λ演算都被证明了是不朽的模拟计算方法，Turing的文章还是因为勉强打败了Church提出的解决方案被出版。

《通信的数学理论(1948)》

我清楚得记得还在念大学的时候读到这篇论文时的震撼。我之前认为信息就跟幽默感和美感一样，是无形的。这篇论文让信息突然变成了准确而可量化的了。

Claude E. Shannon是信息论的鼻祖，他让现代信息和通讯技术成为了可能。

《非合作博弈(1950)》

这篇并不是跟计算相关的，严格意义上它甚至都不能算作一篇论文。John Nash的博士学位论文整整26页却只有2个引用，其中一个还是引用他自己以前的一篇论文。在这篇文章中，Nash阐述了一个基于非合作博弈的数学理论。这个理论在经济学和冷战策略上有着深远的影响，并且最终带给了Nash诺贝尔经济学奖。

就算你仅仅想看看Nash因为无法用当时的打字机打出来而手写的那些数学符号，这篇论文已经很值得一读了。

《递归函数下的符号表达式及计算机器(第一部分)(1960)》

给出John McCarthy)这篇Lisp的开山之作不足为奇。这篇文章把编程原理讲解地细致到了原子级别。

Carl Sagan说过要想完完全全从零开始做一个苹果派，你先得创造一个宇宙。McCarthy在这篇文章告诉我们，要想完完全全从零开始做计算，你只需要先创造S表达式。

《分布式系统中的时间，时钟和事件顺序(1978)》

Leslie Lamport在这篇文章里研究了同步和因果关系的极限，可以说能让人想起爱因斯坦的相对论。虽然他的文章在互联网出现之前就有了（论文参考了ARPA net），但是这篇论文开始谈及到与现代分布式系统密切相关的协调问题。

Lamport在2013年获得了图灵奖。有个笑话是：每个人都知道他早就该得图灵奖了，但大家用了很长时间才达成了“同步”一致。

《概率加密(1983)》

现代加密方法往往依赖于陷门函数 —— 它们的计算量不大，但几乎无法被逆推。这使得人们可以花费很小的代价加密数据，并且可以很有信心的说这是不可逆转的过程，而且在不知道关键信息的情况下解密数据是不可能的。RSA非对称密码体系)就是一个著名的例子。

但这类方法有个漏洞，即它们仍然允许密码破译者获得加密数据的部分信息。在这篇文章中， Shafi Goldwasser和Silvio Micali提出且证明了一种弥补这个漏洞的方法——用随机的方式使加密过程不确定。

《编写自成长的语言（1998）》

编写代码是非常有意思的，开发一门语言同样如此。在这篇论文中，Guy Steele一开始让单词只有一种意思，然后再让单词同时具有多种意思。他提出了一个很好的观点，成长是一门语言的关键，其能力应该能随着时间的推移而改变，在人们编写代码的时候就能够诞生新的特性。

## 《实用的拜占庭容错和实时恢复（2002）》

互联网是个很可怕的地方。网络和服务不仅可能完全无法使用，有时也会出现各种奇怪或恶意的行为，甚至互相损害。

做一个可容错的分布式系统常常会被称为“拜占庭将军难题”——意思是假想好几只军队需要协同合作才能成功攻城，但领导他们的将军们却并不完全可靠。

在这篇文章里，Miguel Castro和Barbara Liskov描述了一个很有效率的算法，来保证即使三分之一的节点以任何恶意的形式失效，系统仍然是健康的。

对今天的分布式系统而言，这类算法非常有用。比如比特币系统，用了这种算法它只要没失去一半以上的计算能力，都可保证可用。

## 《命题类型（2014）》

Philip Wadler的任何一篇论文都可以上榜。这篇论文在我写作的时候仍然是草稿，却可谓是新的经典之作，论文中Walder的引用范围从电影独立日到乐团双杰，他揭露了Curry-Howard同构性这种精妙对称性的本质。

同构性记录了逻辑命题和函数式程序种类之间非同寻常的关系。更重要的是，论文的开篇引用了Brouwer在1923年的论文中提出的直觉主义，作为函数对应于传统的直觉主义的构造性证明，这篇论文验证了直觉主义的重要性。

本文在我心中占有特殊的地位，因为在Wadlerd发布最新起草的文章的那个早上，我正好在做关于Curry-Howard同构性的演讲。但幸运的是，其内容没有什么实质性的改变，因而我没有给观众过时的信息。）

## 总结

我开始写这篇文章的时候正在抱怨我所在行业的短视。尽管我们这个行业的历史不长，从业者仍然对历史很不熟悉。我之前听说，有人问Bjarne Stroustrup，把lambda表达式引入C++是不是因为lambda表达式是个新潮的玩意儿——lambda演算可以追溯到20世纪30年代，比C++要年纪大多了。

但当回顾这份清单上的计算机科学家们的背景时，我不得不为自己的短视感到惊讶。上面说的这12个科学家中，有10个都是男性，且为白人。

因此，我想总结说，除非我有严重的偏见，我们仅仅用到了人类才华总和中很小的一部分。当我思考过去百年我们在计算机科学里走了多远的时候，我无法想象如果能调用更多的聪明才智，下个百年我们会有多么伟大的成就。

ThoughtWorks

# *100 years of*

# COMPUTER SCIENCE

10 Academic Papers that Changed Computing Over 10 Decades

**1923**

On the significance of the principle of excluded middle in mathematics, especially in function theory

***Luitzen Egbertus Jan Brouwer***

On Countable Numbers, with an Application to the Entscheidungs problem

***Alan Turing***

**1936**

**1948**

A Mathematical Theory of Communication

***Claude E. Shannon***

Non-Cooperative Games
***John Nash***

**1950**

**1960**

Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part 1
***John McCarthy***

Time, Clocks, and the Ordering of Events in a Distributed System
***Leslie Lamport***

**1978**

**1983**

Probabilistic Encryption
***Shafi Goldwasser, Silvio Micali***

Growing a Language
***Guy Steele***

**1998**

2002

Practical Byzantine Fault Tolerance
and Proactive Recovery
***Miguel Castro, Barbara Liskov***

Propositions as Types
***Philip Wadler***

2014

/ThoughtWorks

@ThoughtWorks @ctford

/+thoughtworks

/user/ThoughtWorks

/company/thoughtworks

ThoughtWorks®

creative commons

加入伯乐在线专栏作者。扩大知名度，还能得赞赏！详见《招募专栏作者》

1 赞 6 收藏 评论

## 关于作者：ThoughtWorks

ThoughtWorks是一家全球IT咨询公司，追求卓越软件质量，为客户提供注重实效的咨询服务（IT组织优化、技术咨询、测试策略、客户体验）。同时我们也擅长构建定制化系统和产品，帮助客户快速将概念转化为价值。 我们满怀信心和激情，不断推动敏捷在中国IT业的推广和应用，提升中国软件开发的生产力。

合作联系

Email：bd@Jobbole.com

QQ： 2302462408 （加好友请注明来意）

## 更多频道

小组 - 好的话题、有启发的回复、值得信赖的圈子
头条 - 分享和发现有价值的内容与观点
相亲 - 为IT单身男女服务的征婚传播平台
资源 - 优秀的工具资源导航
翻译 - 翻译传播优秀的外文文章
文章 - 国内外的精选文章
设计 - UI,网页，交互和用户体验
iOS - 专注iOS技术分享
安卓 - 专注Android技术分享
前端 - JavaScript, HTML5, CSS
Java - 专注Java技术分享
Python - 专注Python技术分享