# 计算机科学概述

信息学院
陆嘉恒
www.jiahenglu.net

# 复习

- 唐纳德 克努特
  - 计算机程序设计艺术
  - Tex编辑软件

- 罗伯特 陶尔扬
  - 数据结构和算法
  - 图的遍历

# Chapter 5

- System Software
- Operating System
  - Function
  - Features
  - Categories
  - Windows
  - Mac OS
  - Unix and Linux

# Chapter 5

- Utilities
  - Windows Utilities
  - Utilities Suites
- Device drivers
- Virus protection and Internet Security
- Careers in IT: Computer Support Specialist

- Homework: pages142: Crossword puzzle, Multiple Choice, Matching

# Chapter 6

- System Unit
- Electronic Data and Instructions
  - Binary Coding Schemes
- System Board
- Microprocessor
- Memory
  - RAM,ROM,CMOS
- System clock

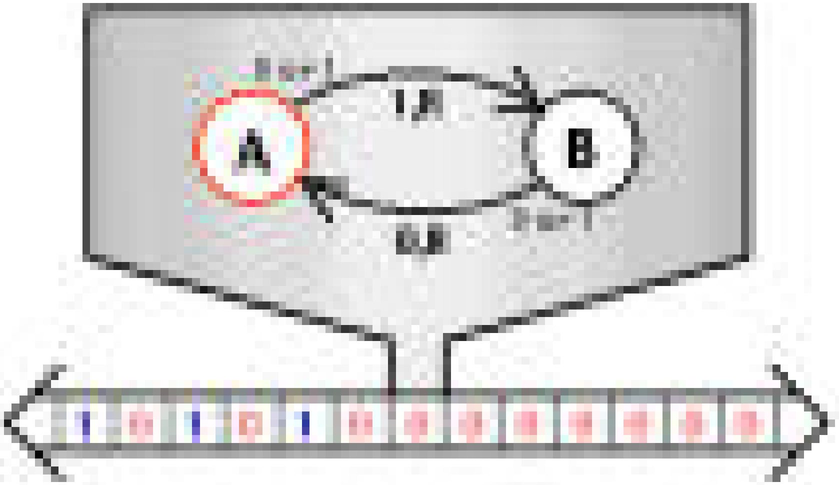
# Chapter 6

- Expansion slots and Card
- Bus lines
- Ports
  - Standard Ports
  - Specialized Ports
  - Cables
- Power Supply
- Careers in IT: Computer Technician
- Homework: pages173: Crossword puzzle, Multiple Choice, Matching

- 艾伦·麦席森·图灵
- Alan Mathison Turing
- 1912年6月23日—1954年6月7日
- 英国数学家

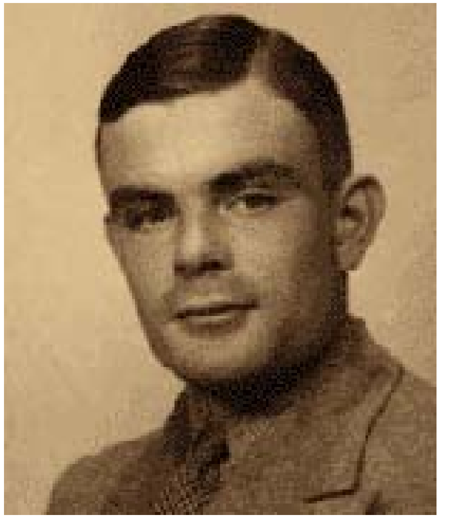
- 1931年图灵进入剑桥大学国王学院，毕业后到美国普林斯顿大学攻读博士学位，二战爆发后回到剑桥。

- **Alan Turing 在二次大战时设计，破解了德国的Enigma密码**



- 1936年，24岁发表著名论文《论可计算数及其在密码问题的应用》，提出了“理想计算机”，后人称之为“图灵机”。图林通过数学证明得出理论上存在“通用图林机”，这为可计算性的概念提供了严格的数学定义，

- 1950年的另一篇著名论文《*Computing Machinery and Intelligence*》 提出了“计算机能思考吗？”（Can Machine Think?），对计算机的人工智能进行了探索，并设计了著名的“图灵测验”。至今，每年都有试验的比赛。



# 图灵测试

- 问： 请给我写出有关“第四号桥”主题的十四行诗。
- 答：不要问我这道题，我从来不会写诗。
- 问：34957加70764等于多少？
- 答：（停30秒后）105721
- 问：你会下国际象棋吗？
- 答：是的。
- 问：我在我的K1处有棋子K；你仅在K6处有棋子K，在R1处有棋子R。现在轮到你走，你应该下那步棋？
- 答：（停15秒钟后）棋子R走到R8处，将军！

# 图灵测试（1）

- 问：你会下国际象棋吗？
- 答：是的。
- 问：你会下国际象棋吗？
- 答：是的。
- 问：请再次回答，你会下国际象棋吗？
- 答：是的。

# 图灵测试 （2）

- 问：  你会下国际象棋吗？
- 答：是的。
- 问：你会下国际象棋吗？
- 答：是的，我不是已经说过了吗？
- 问：请再次回答，你会下国际象棋吗？
- 答：你烦不烦，干嘛老提同样的问题。

- 1954年图灵英年早逝，年仅42岁。

- **图灵奖：**是美国计算机协会（Association for Computing Machinery,ACM）于1966年设立的。

- 图灵奖对获奖者的要求极高，评奖程序极严，一般每年只奖励一名计算机科学家，只有极少数年度有两名在同一方向上做出贡献的科学家同时获奖。

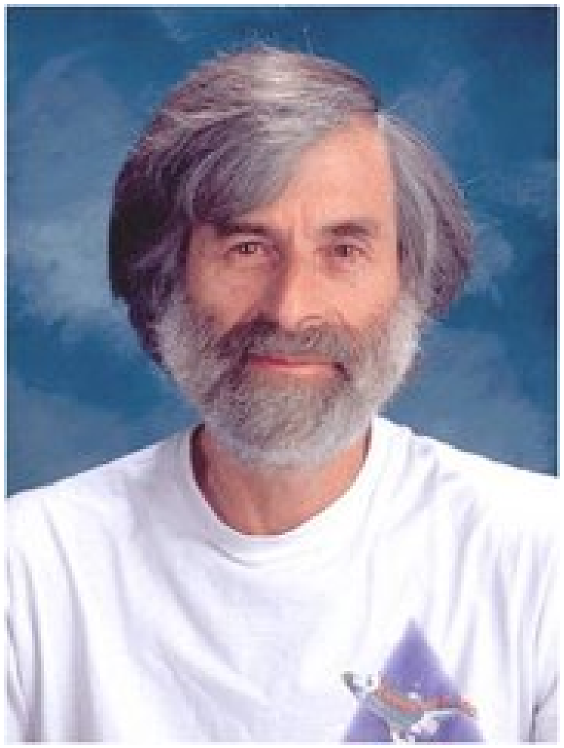
- 因此，尽管“图灵”的奖金数额不算高，但它却是计算机界最负盛名的奖项，有“计算机界诺贝尔奖”之称。

# 2008 图灵奖

- Edmund M. Clarke，Allen Emerson和Joseph Sifakis在开发自动化方法检测计算机硬件和软件中的设计错误上的贡献而授予此计算机行业的最重要奖项。

  模型检测（Model Checking）是一类“验证”，分析设计背后的逻辑，就像数学家用证明来判断一个定理是否正确。

# 莱斯利·兰波特（Leslie Lamport）



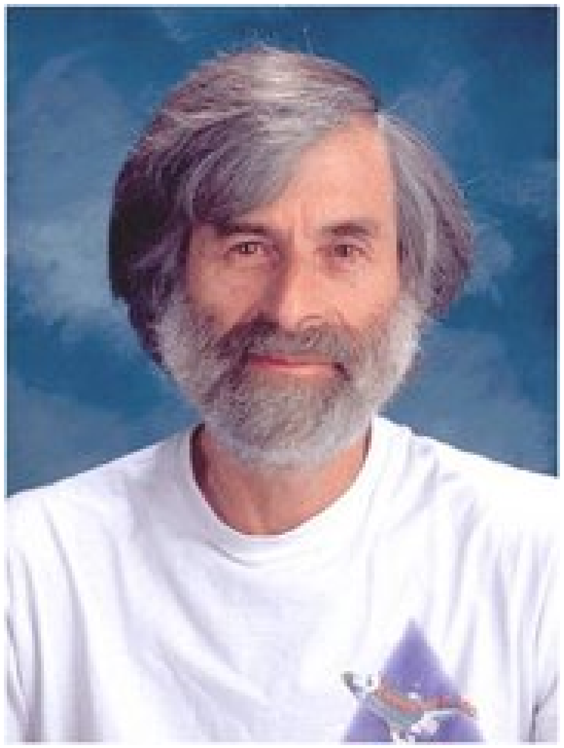
1941年 Leslie　Lamport 生于纽约。

1960年毕业于麻省理工学院数学专业。

1963年获得布兰迪斯大学数学硕士学位。

1965-1969年任教于马尔波罗学院

# 莱斯利·兰波特（Leslie Lamport）



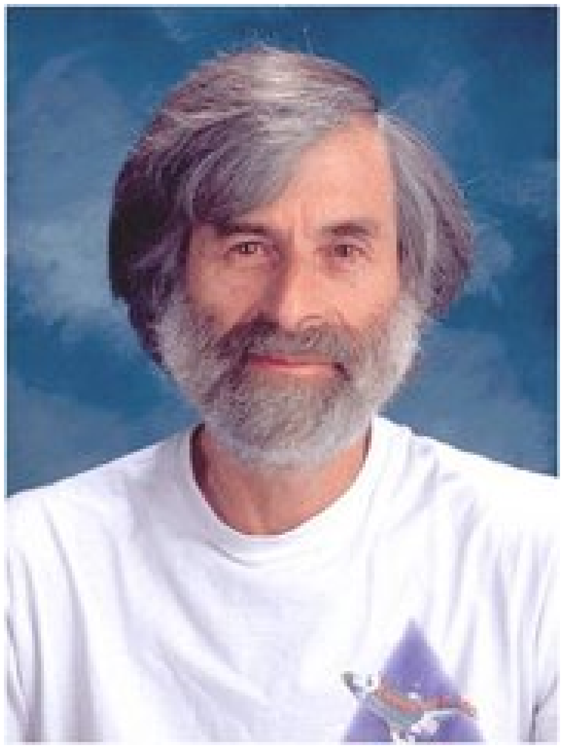
1970-1972年，麻省计算机协会系统设计员。

1972年获得布兰迪斯大学数学博士学位。

1972-1977年，麻省计算机协会研究员。

1977-1985年，SRI 公司计算机科学实验室任究员。

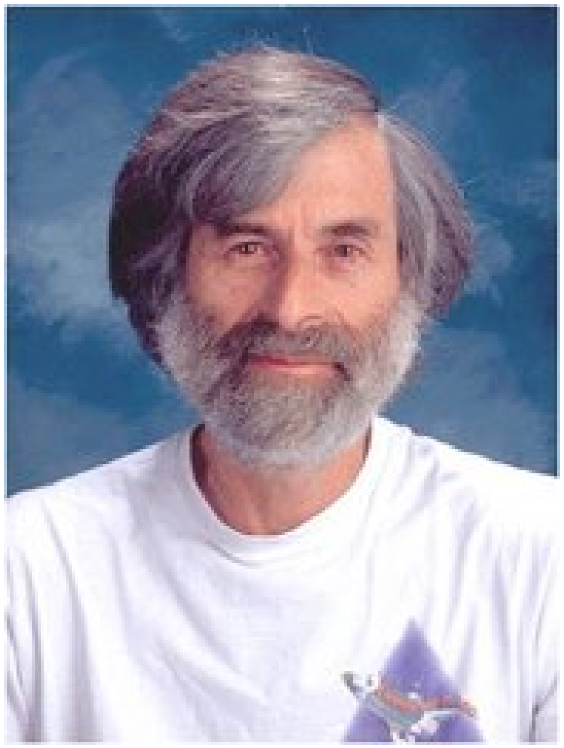
# 莱斯利·兰波特（ Leslie Lamport ）



1982年与另两人共同发表论文“拜占廷将军问题”，既允许军中可能有叛徒，又要保证战争胜利，引申到计算机领域，成为一种容错理论。

1984年前后，编写了便于自己使用的宏包套件，并命名LaTeX。

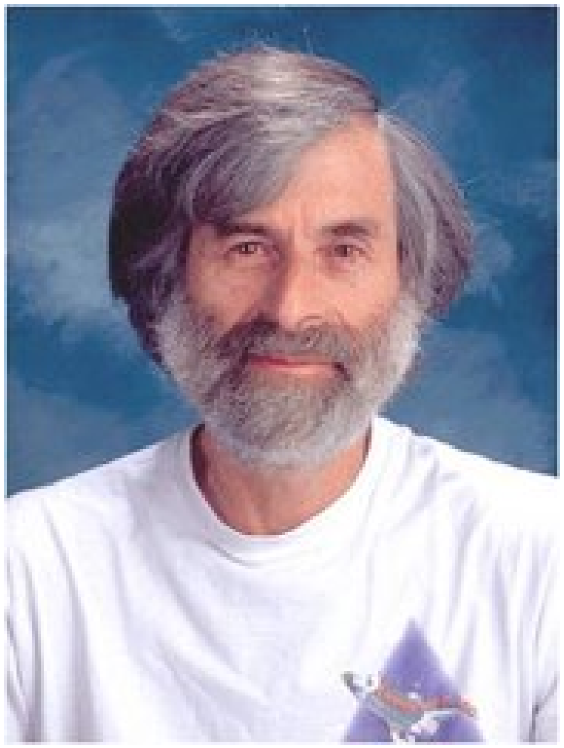
# 莱斯利·兰波特（Leslie Lamport）



1984年发表论文“分布系统中的时间、时钟和事件排序”。

1985-2001年，在数字设备公司以及康柏系统研究中心作研究工作。

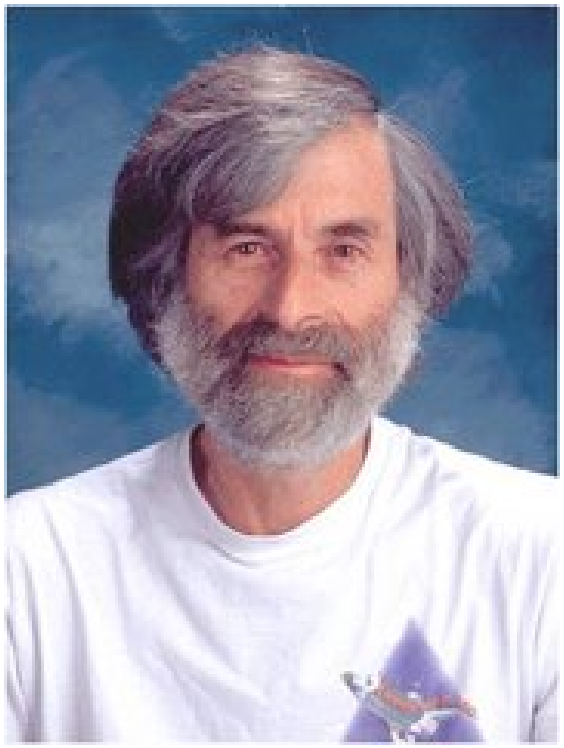
# 莱斯利·兰波特（ Leslie Lamport ）



1989年8月21日，在斯坦福大学 TeX 用户组织会议上，同意将 LaTeX 的维护和开发工作交给 LaTeX3 小组。

1994 年，与 LaTeX3 小组对 LaTeX 作了一次重大改进，版本命名为 LaTeX2e，并出版 LaTeX 使用手册 第二版。

# 莱斯利·兰波特（Leslie Lamport）

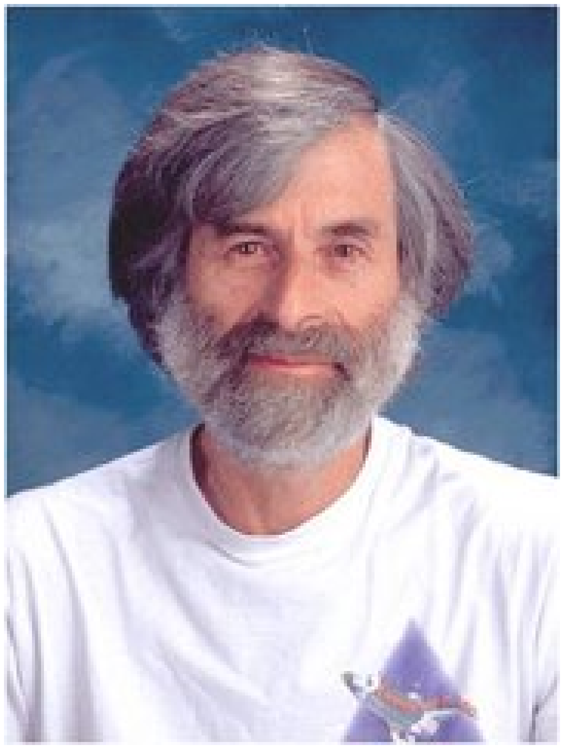


2001年进入位于加利福尼亚的微软研究院，任高级研究员，从事分布式计算机系统理论研究。

2003年7月10日，获德国基尔大学荣誉博士并作学术报告，体恤衫上印着：

“You want proof？I'll give you proof !”

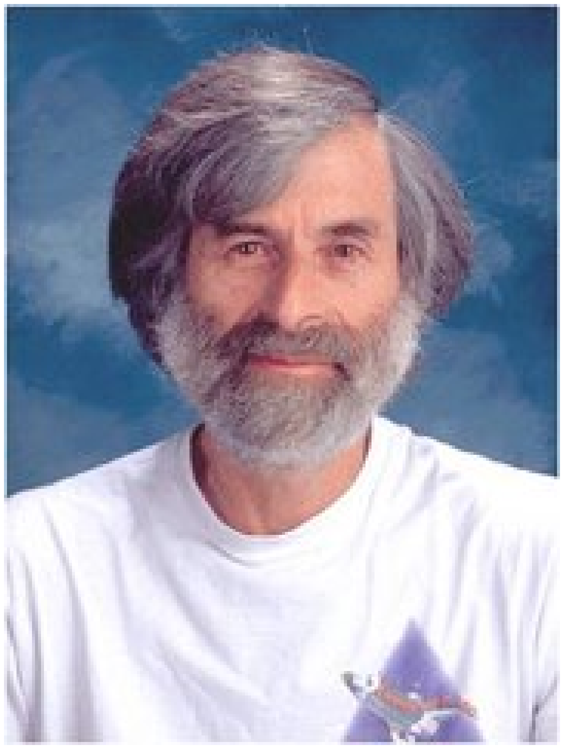
# 莱斯利·兰波特（ Leslie Lamport ）



2004年，由于在计算机信息处理方面的突出贡献，获得皮奥尔奖。同年获瑞士洛桑联邦工业大学荣誉博士。

截至2006年，已发表论文160篇。

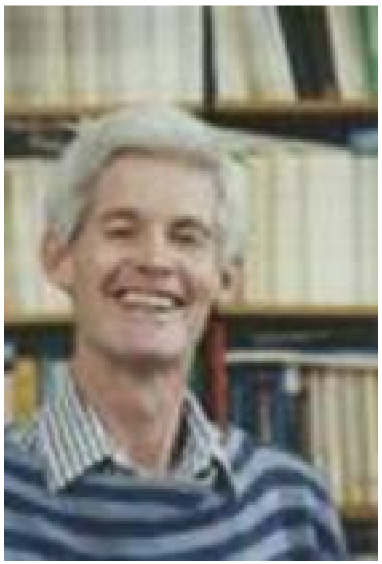
# 莱斯利·兰波特（ Leslie Lamport ）



2004年，由于在计算机信息处理方面的突出贡献，获得皮奥尔奖。同年获瑞士洛桑联邦工业大学荣誉博士。

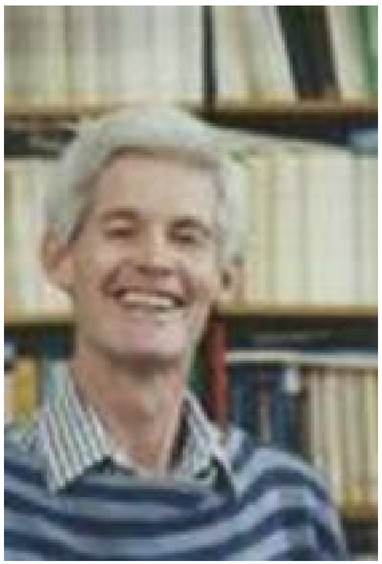
截至2006年，已发表论文160篇。

# 斯蒂夫·库克 （**Stephen Arthur Cook**）



加拿大多伦多大学教授斯蒂芬·库克因在计算复杂性理论方面的贡献，尤其是在奠定NP完全性理论基础上的突出贡献而荣获1982年度的图灵奖。

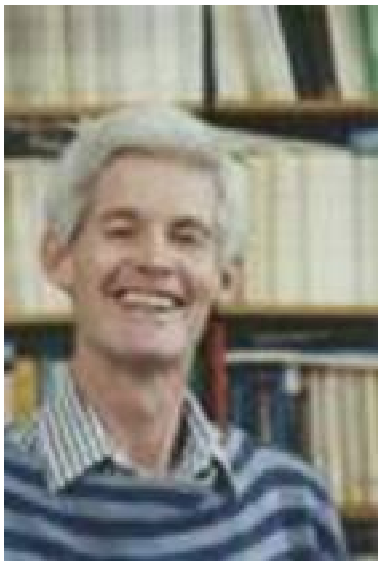
# 斯蒂夫·库克 （**Stephen Arthur Cook** ）



1939年12月14日生于纽约州的布法罗(Buffalo)

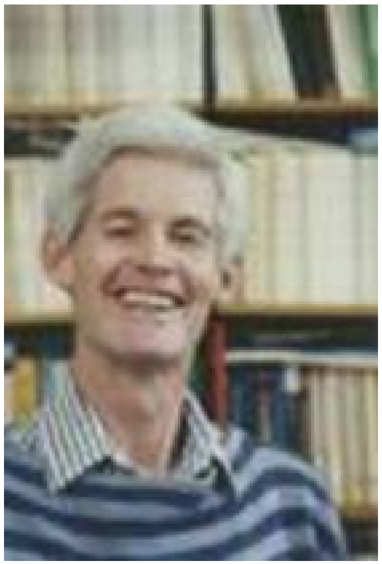
1957年中学毕业后，库克离开克拉伦斯去上密歇根大学。

# 斯蒂夫·库克 （**Stephen Arthur Cook** ）



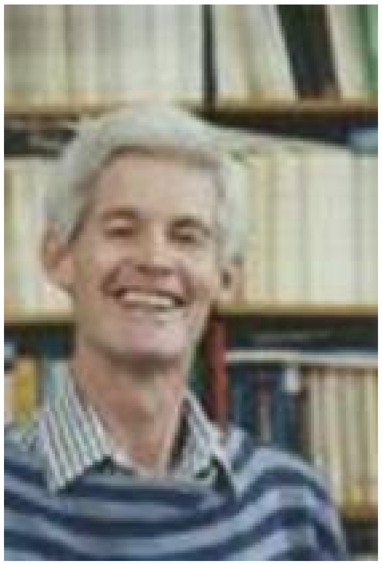
编了一个程序以验证哥德巴赫猜想，在机器允许的范围内，每个大于3的偶数都是2个素数之和。

# 斯蒂夫·库克 （**Stephen Arthur Cook** ）



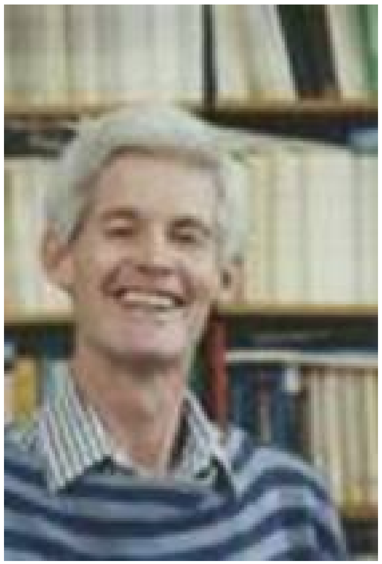
1961年库克获得学士学位以后，转入哈佛大学研究生院深造，第二年就取得了理科硕士学位。他接着攻读数学博士学位

# 斯蒂夫·库克 （Stephen Arthur Cook ）



哈佛研究生院对新兴学科十分重视，虽然计算复杂性理论这一学科分支其时还处于萌芽与初创时期，它就邀请了这方面的一些先驱与奠基人，其中包括拉宾 、哈特马尼斯(J．Hartmanis)和斯坦恩斯(R．Steams) 等 人前来讲学或作报告。

# 斯蒂夫·库克 （**Stephen Arthur Cook** ）



1971年5月，发表了那篇著名的论文："定理证明过程的复杂性"(The Complexity of Theorem Proving Procedures)，在这篇论文中，库克首次明确提出了NP完全性问题

# NP完全性问题

$$2^{67}-1= a \times b \ ?$$

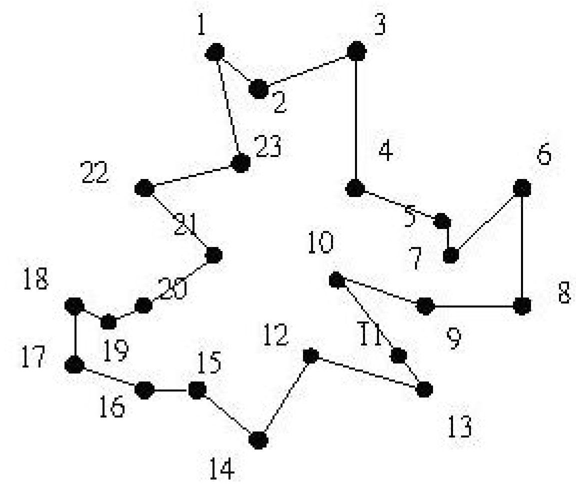
$$2^{67}-1= 193{,}707{,}721 \times 761{,}838{,}257{,}287$$

# NP完全性问题

- 千僖年数学难题（2000-5-24，美国的克雷(Clay)数学研究所，在巴黎法兰西学院宣布每一个悬赏一百万美元）
  http://www.claymath.org/millennium

  - 之一： P (多项式算法)问题对NP (非多项式算法)问题
  - 之二： 霍奇(Hodge)猜想
  - 之三： 庞加莱(Poincare)猜想
  - 之四： 黎曼(Riemann)假设
  - 之五：。。。

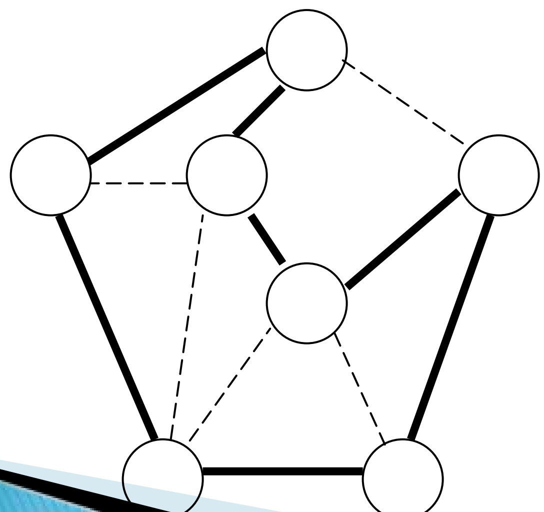
# NP完全性问题

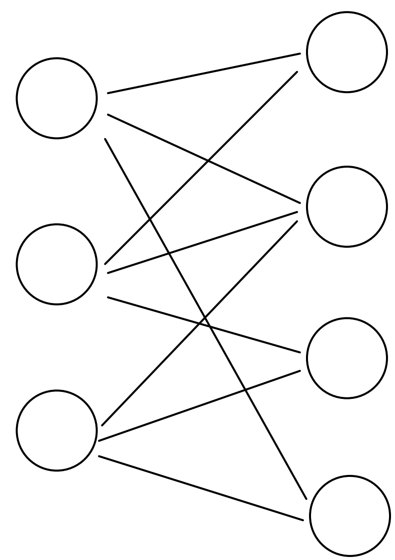



- 多项式时间算法，简单！
- 非多项式时间算法，复杂！
- 找不到可解算法的问题，复杂！
- 旅行商问题（**Traveling Salesman Problem**）：并回到原出发点的最短路线
  - 1→2→3(1→3→2) → …… , 这两种走法的距离不一样。
  - 可能的路线： n !
  - 怎样找出总路程最短(省钱)的一种走法？穷举法！没有最佳的方法。
  - NP完全问题
  - 应用：运输公司配送货物；邮递员；生产线上组装工序；……
  - 穷举法：如果 T(20) = 1 h, T(21) = 21 h, … , T(25) = 728 y

# NP完全性问题

- **Non-Deterministic Problem(无法按部就班直接地计算出来，e.g.)**
  - **Factorization（大的合数分解质因数的问题）：M = ？ × ？**
  - **A Hamiltonian cycle of a graph *G* is a cycle that visits each vertex of the graph exactly once.**
  - **Problem: Given a graph *G*, does *G* have a Hamiltonian cycle?**





# NP完全性问题

- **Problems in P can be solved "quickly"**
- **Problems in NP can be verified "quickly"**
- **It is easier to verify a solution than to solve a problem.**

**In 1903, F. N. Cole factored $2^{67}-1$, i.e.**

$$2^{67}-1 = 761{,}838{,}257{,}287 \times 193{,}707{,}721$$

**It took Cole 150 days to find the factorization! but once it was found, we can verify it very quickly!**