# 数据与算法的故事: 1

吕正华

2019.9

# 自我介绍

- Pivotal 资深软件工程师，开发 Greenplum 内核
- kainwen@gmail.com
- 个人主页: https://kainwen.com
- 2014 年毕业于电子系智能感知实验室, 工学硕士
- 2012 年第一次担任数据与算法助教

# 今天的话题

故事安排总览

资料和工具推荐

今天话题: 算法分析

结论

# 数据与算法 2019 年秋季学期故事

- Z 变换, Analytic Combinatorics 和算法分析
- 斐波那契数列的故事：矩阵连乘和数论基础
- 随机分析与一致性哈希算法
- 搜索算法与 Prolog 编程语言
- 形式化证明初探
- 矩阵相关的故事

# Z 变换, Analytic Combinatorics 和算法分析

*People who analyze algorithms have double happiness. First of all they experience the sheer beauty of elegant mathematical patterns that surround elegant computational procedures. Then they receive a practical payoff when their theories make it possible to get other jobs done more quickly and more economically.*

***D. E. Knuth***

# 算法分析

- $\mathcal{O}$ 符号只能描述算法最坏的情况
- 仅仅知道 $\mathcal{O}$ 符号表述的复杂度无法比较算法快慢
- 我们需要对算法的实现精准建模，精准求解关键步骤的公式
- 生成函数和渐进逼近技术 (电子系数字信号处理课程的 z 变换和系统分析)
- 这部分是今天的话题，在后面 section 会附上非常多的例子

# 斐波那契数列的故事：矩阵连乘和数论基础

- 著名的斐波那契数列 $F_n = F_{n-1} + F_{n-2}, F_0 = 0, F_1 = 1$
- 泰勒展开 $\frac{z}{1-z-z^2}$ 能发现什么？
- 曾经 Online Judge 一个练手题：编程求解 $F_n$，结果模 $10007$

# 搜索算法与 Prolog 编程语言

- 深度优先搜索和广度优先搜索的本质区别
- 编程语言的求值顺序模型: 正则序和应用序
- Prolog 语言及其应用
- HackMem 里的游戏: 孔明棋

# 形式化证明初探

*I'm aware that many students don't see the importance of a mathematical approach to CS. The feeling is, just let me near a keyboard and let me code. It's quite common.*

***Jeffrey Ullman***

- Linus: *Talk is cheap, show me the code.*
- 某知乎用户: *Code is cheap, show me the proof.*
- 《C 专家编程》作者关于形式化验证的论断在今天还试用么?
- 证明和编程是等价的，写数学证明就是在编程
- 理解证明对设计算法和理解算法很重要

# 矩阵相关的故事

还没有想好:)

# 编程相关的工具

- 文本编辑器: Emacs, Vim, Atom, VS Code, SublimeText, ...
- IDE: Clion, Eclipse, Visual Studio, ...
- 代码阅读工具: Clion, Understand, Eclipse, CScope+Emacs, ...
- 调试工具: gdb 和其他 IDE 自带工具
- 其他: Linux, Git, Make, Perf, Dtrace, ...

# 书籍推荐

- 具体数学
- 算法导论
- 算法分析导论
- C Interfaces and Implementations
- 深入理解计算机系统
- Structure and Interpretation of Computer Programs
- 算法
- HACKMEM
- 黑客: 计算机革命的英雄

# 例子 1: 两千多年前的算法 GCD

**Result:** GCD(a, b)
**if** $b == 0$ **then**
 | Return a
**else**
 | GCD(b, a % b)
**end**

**Algorithm 1:** 欧几里得辗转相除算法求最大公约数

- 状态转移序列 $(a, b) -> (b, a\%b) \ldots (*, 0)$
- 最坏情况复杂度分析: $F_{n+2} \geq F_{n+1} + F_n$
- $\mathcal{O}(log(b))$

# 例子 2: 快速排序划分算法交换次数

```
1   private void quicksort(int[] a, int lo, int hi)
2   {
3           if (hi <= lo) return;
4           int i = lo-1, j = hi;
5           int t, v = a[hi];
6           while (true)
7           {
8              while (a[++i] < v) ;
9              while (v < a[--j]) if (j == lo) break;
10             if (i >= j) break;
11             t = a[i]; a[i] = a[j]; a[j] = t;
12          }
13          t = a[i]; a[i] = a[hi]; a[hi] = t;
14          quicksort(a, lo, i-1);
15          quicksort(a, i+1, hi);
16  }
```

- 6 到 12 行的代码里，第 11 行执行的平均次数是多少？
- 数据是无重复的均匀分布，数组长度是 $N$, 则 $\frac{N-2}{6}$

# 例子 3: Coupon Collector 问题 1

- 网易云音乐的一个歌单有 N 首歌，随机听，需要听多少首，才能恰好每首歌都听到？
- 分布式数据库两阶段哈希聚合操作的查询计划，估计第一个阶段 spill 到磁盘的数据量
- 可以用生成函数的技巧，也可以用期望的线性性质技巧（有限状态机）
- 介绍一下我的技巧

# 例子 3: Coupon Collector 问题 2

- 技巧: $E(n) = \sum_{N \geq 0} Pr\{n > N\}$
- 任务转换成推理: $Pr\{$ 听了 $k$ 首歌还没有搞定 $\}$
- 任务转换为建模事件 $A = \{$ 听了 $k$ 首歌还没有搞定 $\}$
- 记录事件 $B_i = \{$ 听了 $k$ 首歌还缺 $i\}$
- $A = B_1 \cup B_2 \cup \cdots \cup B_N$
- 利用容斥原理可以得到公式
  $\sum_{k \geq 0} \sum_{m=1}^{N} (-1)^{m+1} \binom{N}{m} (\frac{N-m}{N})^k$

# 例子 3: Coupon Collector 应用于分布式数据库

- `select c1, sum(c2) from t group by c1`
- 分布式表 t 按 c3 分布
- 类似思考，全电子系学生是一个表，c1 是省份，c2 是学号，c3 是班级号, 计算电子系学生按省划分的总学号
- 可以用两阶段分布式计算，第一个阶段利用哈希聚集
- 数量太大的时候，要利用外存计算，外存会产生可观的代价, 必须估计写入外存的数据量
- 简单抽象, $1, 2, 3, 4, ..., 100$ 个值，每个值有 $10$ 个重复, 纯随机，平均取多少个数才能取到 $40$ 个不同的数值？
- 实际产品的代码: https://github.com/greenplum-db/gpdb/pull/8439

# 总结

数据和算法课程是至关重要的基础课，且和电子系其他学科密不不可分。它是考察综合素质的绝佳课程。

- 严谨: 程序算法的正确性，需要严格的证明 (离散数学和数理理逻辑)
- 效率: 算法性能的分析，必须非常严格精确才有意义 (　生成函数、Z 变换、算法)
- 工程: 最终交付实现，需要编程，调试和优化 (工具链，操作系统，编译器，编程语言，体系结构)

# 问题

1. 搜索或者自行推理 Coupon Collector 问题的经典解法
2. 搜索阅读欧几里得辗转相除法平均情况的执行步数相关资料
3. 用高中的数学技巧求解
   $C_N = N + 1 + \frac{1}{N}\sum_{1\le j\le N}(C_{j-1} + C_{N-j}), C_0 = 0$
4. 证明 $\sum_{k\ge 0}\sum_{m=1}^{N}(-1)^{m+1}\binom{N}{m}(\frac{N-m}{N})^k = NH_N$
5. 解决不放回的 Coupon Collector Problem

# 参考文献

同推荐书目

# 结束

Q & A ?

谢谢 !