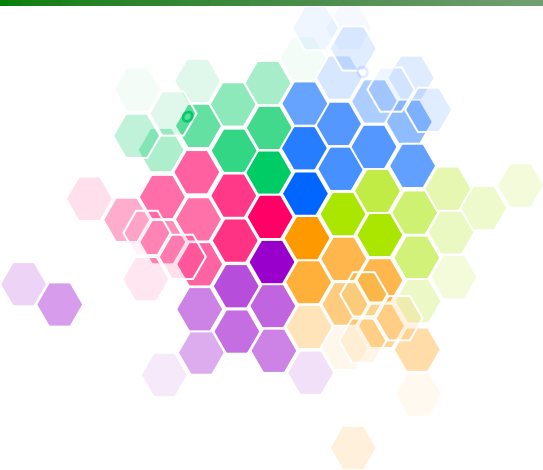


算法与数据结构

主讲：郭龙坤



教材与参考书

- 教材

 - 56 学时

- 参考书

 - Introduction to Algorithms, 中文名: 算法导论, T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein
 - 数据结构与算法, A.V. 阿霍, J.E. 霍普克罗夫特等, 科学出版社
 - The Art of Computer Programming, 中文名: 计算机程序设计艺术, Donald E. Knuth

学时安排与成绩评定

- 学时安排

- 80 = 授课/讨论（56理论）+ 上机（24实践）

- 授课与上机

- 授课：1-14周

- 上机：2-13周

- 成绩评定

- 平时作业+上机作业/考试（40%） + 笔试（60%）

教学目的:

- 学习常用的数据结构，熟悉各种基本数据结构的特点、存储表示、运算方法、典型应用；
- 了解每一种数据结构的使用代价和益处，培养同学根据实际问题的要求，选择合适的数据结构设计算法的能力；
- 掌握一些典型算法，培养算法分析的能力；能够编写解决较为复杂问题的程序。

□ 学科的特点！



面向问题驱动的实践教学模式

- 实验与作业安排
 - 每章一次作业题，2-3道
 - 第2周开始上机编码作业



课程教学网站

网址1: <https://pintia.cn/home>

网址2: <http://ds.fzu.edu.cn>

为什么要学习数据结构与算法

- 用计算机解决实际问题，就是要在计算机中建立一个解决这个问题的模型。
- 程序是使用程序设计语言精确描述的实际问题求解的模型。
- 程序中描述的**数据**用来表示问题中涉及的**对象**，程序中描述的**函数(过程)**表示了了对于数据的处理**算法**；通过接受实际问题的输入，经过程序的运行，便可以得到实际问题的一个解。

为什么要学习数据结构与算法

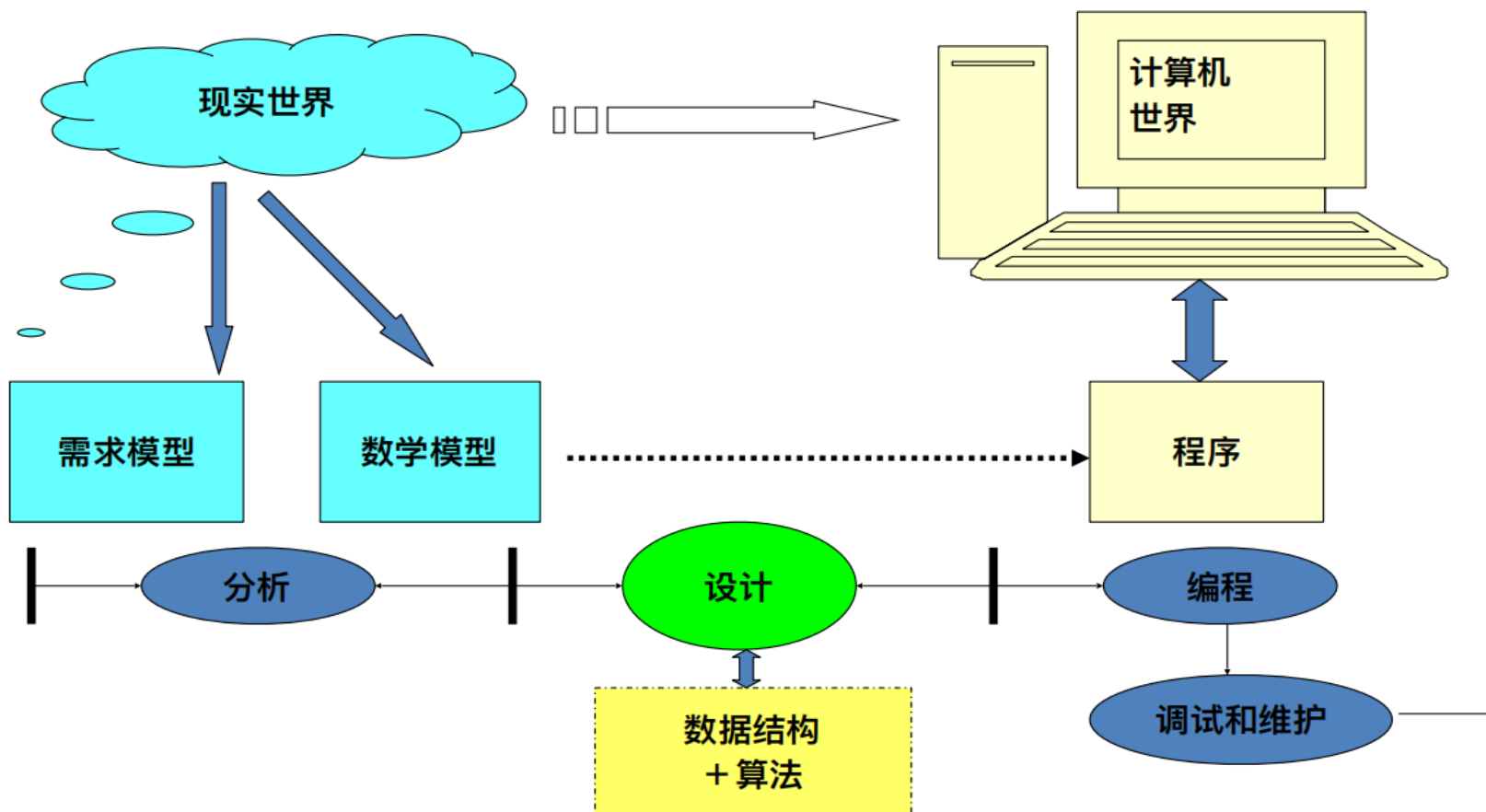
N.沃思 (Niklaus Wirth)教授提出:

$$\text{程序} = \text{算法} + \text{数据结构}$$

以上公式说明了如下两个问题:

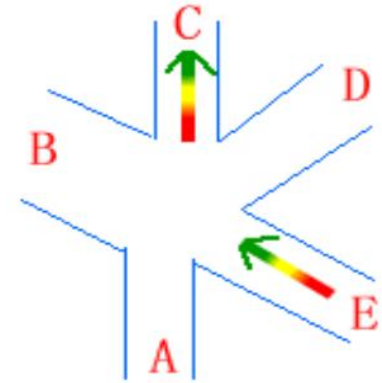
- (1) 数据上的算法决定如何构造和组织数据
(算法→数据结构)。
- (2) 算法的选择依赖于作为基础的数据结构
(数据结构→算法)。

用计算机求解问题



问题求解示例

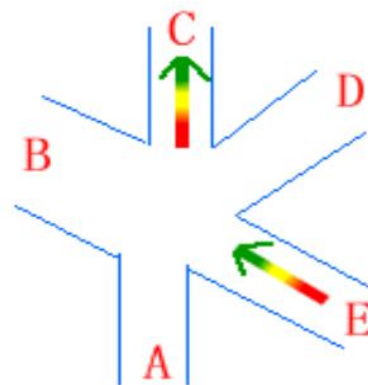
- 为一个多叉路口设计信号灯管理系统
- 对可能行驶路线实行分组：
 - 组内各方向行驶无冲突(可行)
 - 组数尽可能少(有效—最优解)



问题求解示例

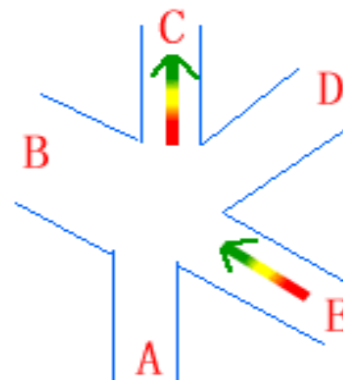
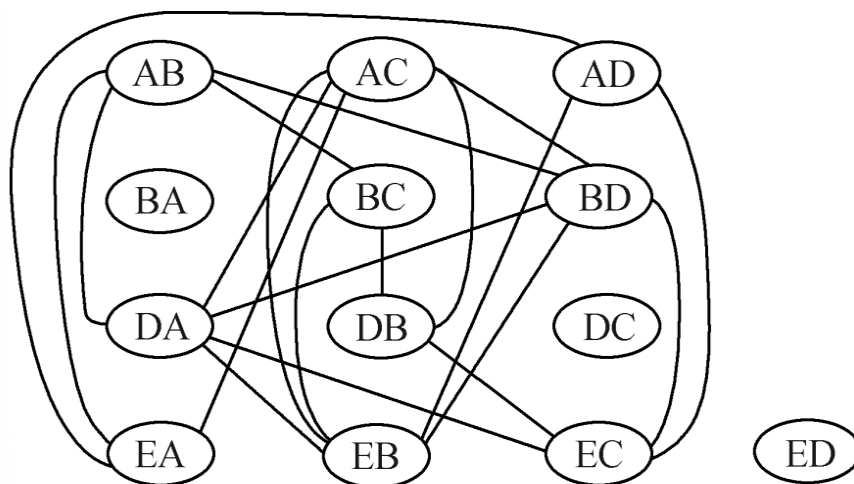
■ 信号灯问题分析

- $A \rightarrow B$, $A \rightarrow C$, $A \rightarrow D$,
- $B \rightarrow A$, $B \rightarrow C$, $B \rightarrow D$,
- $D \rightarrow A$, $D \rightarrow B$, $D \rightarrow C$,
- $E \rightarrow A$, $E \rightarrow B$, $E \rightarrow C$,
 $E \rightarrow D$ 。



问题求解示例

■ 信号灯问题抽象



问题求解示例

■ 着色问题

- 把上图中的一个结点理解为一个国家，结点之间的连线看作两国有共同边界。
- 上述问题就变成著名的“图着色问题”：求出（最少）要几种颜色可将一个地图中所有国家着色，使得任意两个相邻的国家颜色都不相同。

■ 穷举法

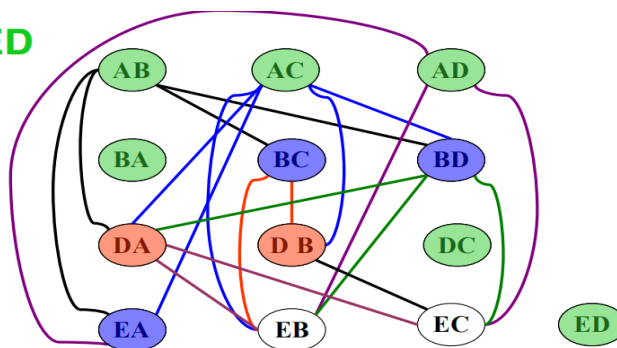
- 从分为1、2、3...组开始考察，逐个列举出所有可能的着色方案，检查这样的分组方案是否满足要求。首先满足要求的分组，自然是问题的最优解。

问题求解示例

■ 贪心法

- 先用一种颜色给尽可能多的结点上色；然后用用另一种颜色在未着色结点中给尽可能多的结点上色；如此反复直到所有结点都着色为止。

- 绿色: **AB AC AD BA DC ED**
- 蓝色: **BC BD EA**
- 红色: **DA DB**
- 白色: **EB EC**



■ 如何写程序实现？？

DS与Algo的前沿科学在做什么？

sigact.org/prizes/best_paper.html

ACM Special Interest Group on Algorithms and Computation Theory

Winners

STOC 2021

- “A (Slightly) Improved Approximation Algorithm for Metric TSP”, by Anna R. Karlin (University of Washington), Nathan Klein (University of Washington), and Shayan Oveis Gharan (University of Washington).
- “The Complexity of Gradient Descent: $CLS = PPAD \cap PLS$ ”, by John Fearnley (University of Liverpool), Paul W. Goldberg (University of Oxford), Alexandros Hollender (University of Oxford), and Rahul Savani (University of Liverpool).
- “Indistinguishability Obfuscation from Well-Founded Assumptions”, by Aayush Jain (University of California at Los Angeles), Huijia Lin (University of Washington), and Amit Sahai (University of California at Los Angeles).

STOC 2020

- “Improved Bounds for The Sunflower Lemma”, by Ryan Alweiss, Shachar Lovett, Kewen Wu, and Jiapeng Zhang.

STOC 2019

- “Log-Concave Polynomials II: High-Dimensional Walks and an FPRAS for Counting Basis of a Matroid”, by Nima Anari, Kuikui Liu, Shayan Oveis Gharan, and Cynthia Vinzant.
- “The Reachability Problem for Petri Nets Is Not Elementary”, by Wojciech Czerwiński, Sławomir Lasota, Ranko Lazić, Jérôme Leroux, and Filip Mazowiecki.
- “Oracle Separation of BPQ and PH”, by Ran Raz and Avishay Tal.

STOC 2018

- “A Constant-Factor Approximation Algorithm for the Asymmetric Traveling Salesman Problem”, by Ola Svensson, Jakub Tarnawski, and László A. Végh.

STOC 2017

- “Explicit, Almost Optimal, Epsilon-Balanced Codes”, by Amnon Ta-Shma.
- “Deciding Parity Games in Quasipolynomial Time”, by Cristian Calude, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, and Frank Stephan.
- “A Weighted Linear Matroid Parity Algorithm”, by Satoru Iwata and Yusuke Kobayashi.

STOC 2016

- “Reed-Muller Codes Achieve Capacity on Erasure Channels”, by Shrinivas Kudekar, Santhosh Kumar, Marco Mondelli, Henry D. Pfister, Eren Sasoglu, and Rudiger Urbanke.
- “Explicit Two-Source Extractors and Resilient Functions”, by Eshan Chattopadhyay and David Zuckerman.
- “Graph Isomorphism in Quasipolynomial Time”, by László Babai.



清华主页 · 清华新闻 · 学术科研 · 正文

交叉信息研究院段然团队获得STOC 2025最佳论文奖

清华新闻网5月13日电 近日，清华大学交叉信息院段然团队的论文“突破有向单源最短路径的排序障碍”（Breaking the Sorting Barrier for Directed Single-Source Shortest Paths）在理论计算机国际顶级会议STOC 2025（ACM SIGACT Symposium on Theory of Computing，ACM理论计算特别兴趣小组理论计算研讨会）上获得最佳论文奖。

段然团队探讨了图论算法中经典的“单源最短路径问题（SSSP）”。Dijkstra算法是解决这一问题的基本算法，它以荷兰计算机科学家埃德斯格尔·迪克斯特拉（Edsger Dijkstra）的名字命名。自1956年开发以来，该算法一直被认为是一项里程碑式的贡献，被广泛应用于导航系统等。为了改进Dijkstra算法，该论文仔细研究了造成其大部分计算成本的部分：与所谓“优先队列”的交互。在执行过程中，Dijkstra算法需要维护“前沿”，即已经发现但尚未完全处理的顶点集合——这意味着它们与源点的暂定最短距离是已知的，但算法可能尚未完全探索它们的邻近顶点。这些前沿顶点存储在一个优先队列中，并从中反复提取下一个最近的顶点。Dijkstra算法中的前沿顶点可以多达“n”，即顶点的数量。每次提取其中一个顶点的操作开销为 $\log(n)$ ，因此总时间为 $n \log(n)$ 。研究团队提出的新算法通过融合Dijkstra算法和Bellman-Ford的教科书算法，以及一种巧妙设计的允许分组插入和提取的数据结构，递归地缩小了所考虑的前沿的大小。因此，操作的总数可以大大减少，从而缩短了运行时间，使整个算法运行得更快。

2024年图灵奖得主罗伯特·塔尔扬（Robert Tarjan）及合作者发表的论文证明了Dijkstra算法对于“最短路径排序问题”的普遍最优性，获得了FOCS 2024的最佳论文奖，受到了《量子杂志》（Quantum Magazine）和量子位等媒体的报道。单源最短路径问题需要找到从一点s到其他所有点的最短路，而Dijkstra算法的副产品是所有点按照从源点的距离排序，也就是说他们证明的是Dijkstra算法对于这个排序问题是普遍最优的（universally optimal）。但是段然团队的新算法避免了整体排序，所以得到了比Dijkstra算法更快的最短路径问题的算法。而且最短路径问题明显更加重要，更快的最短路径算法在理论和实际应用中都有很大意义。

清华大学交叉信息院副教授段然为论文通讯作者，其他作者包括姚班2019届本科毕业生束欣凯，以及姚班毕业生、交叉信息院2022级博士生毛嘉怡和2021级博士生尹龙晖。斯坦福大学2022级博士生毛啸也作出了贡献。

供稿：交叉信息研究院

如何提高学习与实践能力？

- 学好相关课程

- 程序设计、算法与数据结构等

国家精品课程

算法与数据结构

王晓东

- 参与ACM/ICPC竞赛

程序评测系统

OnlineJudge



acm International Collegiate Programming Contest

- 参与各级创新创业项目

- SRTP、省级、国家级



- 参与各类软件设计、创新创业竞赛

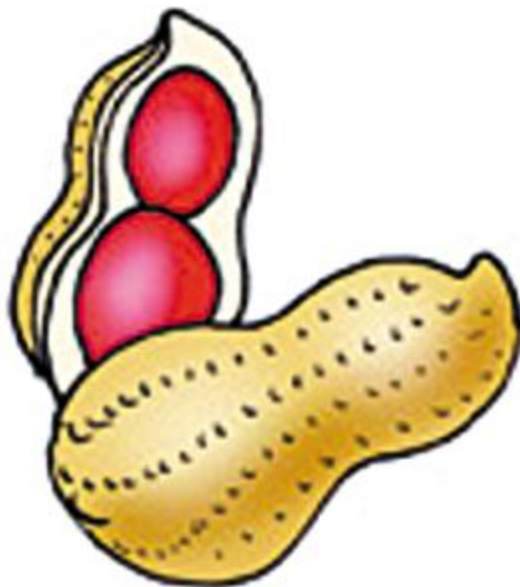
- 省级、国家级



- 参与企业实战项目



花生是果实，也是种子。



天道酬勤--与大家共勉！

- motivation, attitude, passion比智商重要

天道酬勤