



数据结构与算法

第 0 讲：概论

韩文弢

清华大学

2025—2026 学年度春季学期

本讲内容

0.1 问题引入	3
0.2 数据结构	7
0.3 算法	12
0.4 课程信息	21
0.5 本讲小结	24

0.0.1 教学团队介绍

- 授课教师：**韩文弢**
 - 单位：计算机系
 - 开设课程：数据结构与算法、计算思维、程序设计与计算思维、计算机系统概论、程序设计训练
 - 科创指导：学生超算团队、学生算法协会、天格计划团队
 - 邮箱：hanwentao@tsinghua.edu.cn
 - 学术主页：<https://pacman.cs.tsinghua.edu.cn/~hanwentao/>
 - 开放交流时间：单周五 14:00—15:00，双周五 16:00—17:00，提前在 info 预约
- 助教
 - **项晨东**：xcd24@mails.tsinghua.edu.cn
 - **朱悦宁**：zhuyn25@mails.tsinghua.edu.cn

0.1 问题引入

0.1.1 问题 1：图书管理

场景：把一些图书管理起来，要解决如何存放，如何查找的问题。

思考：有哪些方法？

- 方案 1：随便放，查找的时候一本一本翻找。
- 方案 2：根据图书的信息（如领域、书名、作者姓名、出版年份）存放在相应的空间（房间、书架、层等），查找时可以利用有关的索引或顺序。

思考：怎么做效率最高？与什么因素有关？

0.1.2 问题 2：斐波那契数列计算

观察以下两段程序：

```
1 int fib1(int n) {
2     if (n <= 1) {
3         return 1;
4     } else {
5         return fib1(n - 1) + fib1(n -
6             2);
7     }
}
```

```
1 int fib2(int n) {
2     int p = 1;
3     int q = 1;
4     for (int i = 2; i <= n; i++) {
5         int t = q;
6         q += p;
7         p = t;
8     }
9     return q;
10 }
```

思考：它们的效率是否相同？

0.1.3 数据结构与算法

从这些例子可以看到，在不同规模下数据的不同组织方式效率不同，完成同一个计算任务的程序效率可能大不一样。

- 数据结构与算法是侧重研究非数值计算（与方程求解等数值计算相对应）的程序设计问题的领域，目标是设计正确、高效的程序。
- 不仅是复杂程序设计的基础，也是设计和实现编译程序、操作系统、数据库系统及其他系统程序和大型应用程序的重要基础。
- Niklaus Wirth: **Algorithms + Data Structures = Programs**

0.2 数据结构

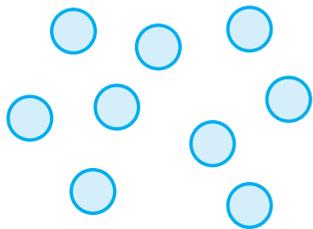
0.2.1 数据结构的定义

定义 0.1 数据结构是指一组具有特定关系的同类数据元素的集合，包含三个要素：

- 数据的**逻辑结构**，是数据在逻辑上存在的关系的总和
- 数据的**存储结构**，是数据实际在计算机中存储的形式
- 数据支持的**操作**（及其实现）

例. 在图书管理的例子中，图书经抽象后就是数据元素，它的属性给出了逻辑结构，它的存放位置是存储结构，需要支持图书的增加、移除、查找等操作。

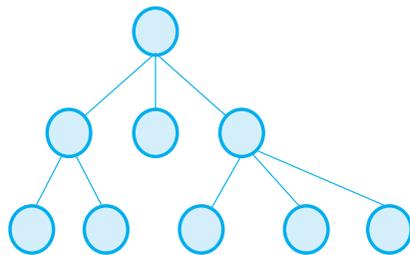
0.2.2 数据的逻辑结构



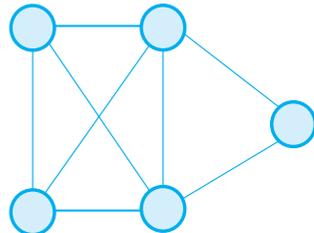
(a) 集合



(b) 线性结构



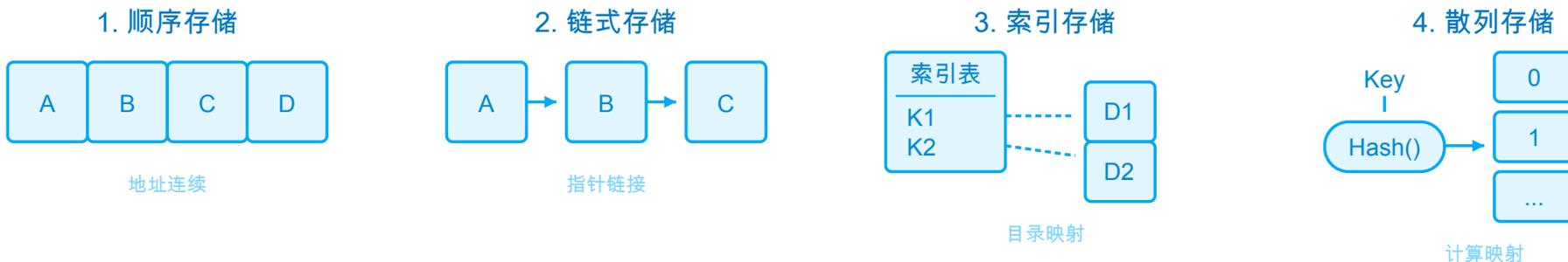
(c) 树形结构



(d) 图结构

- **集合**：所有数据元素之间无顺序关系，只有“是否属于集合”
- **线性结构**：除首尾元素外，每个元素有唯一前驱和唯一后继
- **树形结构**：除根结点外，每个元素有唯一前驱和若干后继
- **图结构**：每个元素都可以有若干前驱和若干后继

0.2.3 数据的存储结构



- **顺序存储:** 存放在连续的存储空间中，存储位置反映逻辑关系
- **链式存储:** 用指针表示逻辑关系，存储上是灵活的
- **索引存储:** 有额外的索引表来加速查找过程
- **散列存储:** 存储位置通过特定的计算得到

0.2.4 数据结构的常见操作

- 创建和释放结构：包括数据结构的创建、初始化，以及必要的释放结构操作。
- 属性操作：包括读取或设置数据结构中各基本属性的值。
- 查找：包括特定搜索、访问和遍历操作。
- 更新：包括插入、删除或修改数据元素的内容，或更新关系。

0.3 算法

0.3.1 算法的词源

中文：“算法”的中文名称出自《周髀算经》，是算经十书中最早的一部。成书不晚于公元前 1 世纪。主要阐述当时的天文学说——盖天说和四分历。书中最早记载了复杂的分数运算、勾股定理和开平方法。

英文：**algorithm** 来自于九世纪波斯数学家花拉子米（al-Khwarizmi）的名字，他从数学角度首先提出了算法这个概念。原为 algorism，意思是阿拉伯数字的运算法则，在 18 世纪演变为 algorithm。

0.3.2 算法的基本概念

定义 0.2 算法是一个有穷规则的集合（有穷的指令集），为解决某一特定类型的问题规定了运算序列。

算法的主要特征：

- **有穷性**：应在执行有穷步后结束
- **确定性**：每一步定义都是确切无歧义的
- **能行性**：每一步运算应足够基本
- **输入**：有零个或多个输入（问题的参数）
- **输出**：有一个或多个输出（计算的结果）

0.3.3 算法举例：辗转相除法求最大公约数

伪码 0.1: 求最大公约数

GCD(x, y):

```
1  | while  $y \neq 0$ 
2  |   |  $t \leftarrow y$ 
3  |   |  $y \leftarrow x \bmod y$ 
4  |   |  $x \leftarrow t$ 
5  | return  $x$ 
```

0.3.4 算法的评价维度

- **正确性**：算法能够按照预定功能产生正确的输出。
- **易读性**：算法逻辑清晰，结构明确，易于阅读、理解和维护。
- **鲁棒性**：算法对于边界条件输入和不频繁出现的输入能够产生正确的输出，对于非法输入能够输出相应提示，而不会出现崩溃。
- **高效性**：算法在时间和空间上高效，并需要较少的运行时间和存储空间。

思考：如何衡量算法的效率？与什么因素有关？

0.3.5 算法效率的评价方式

一般从运行时间和所使用的内存空间两方面来评价算法的效率，分别称为**时间复杂度**和**空间复杂度**。

影响程序实际运行时间的因素有：

- 算法本身的操作步数
- 机器的性能指标
- 编程语言的特点
-

因此理论上在研究算法效率时，一般只考虑其运行时间随着问题规模变大而变大的趋势或程度。

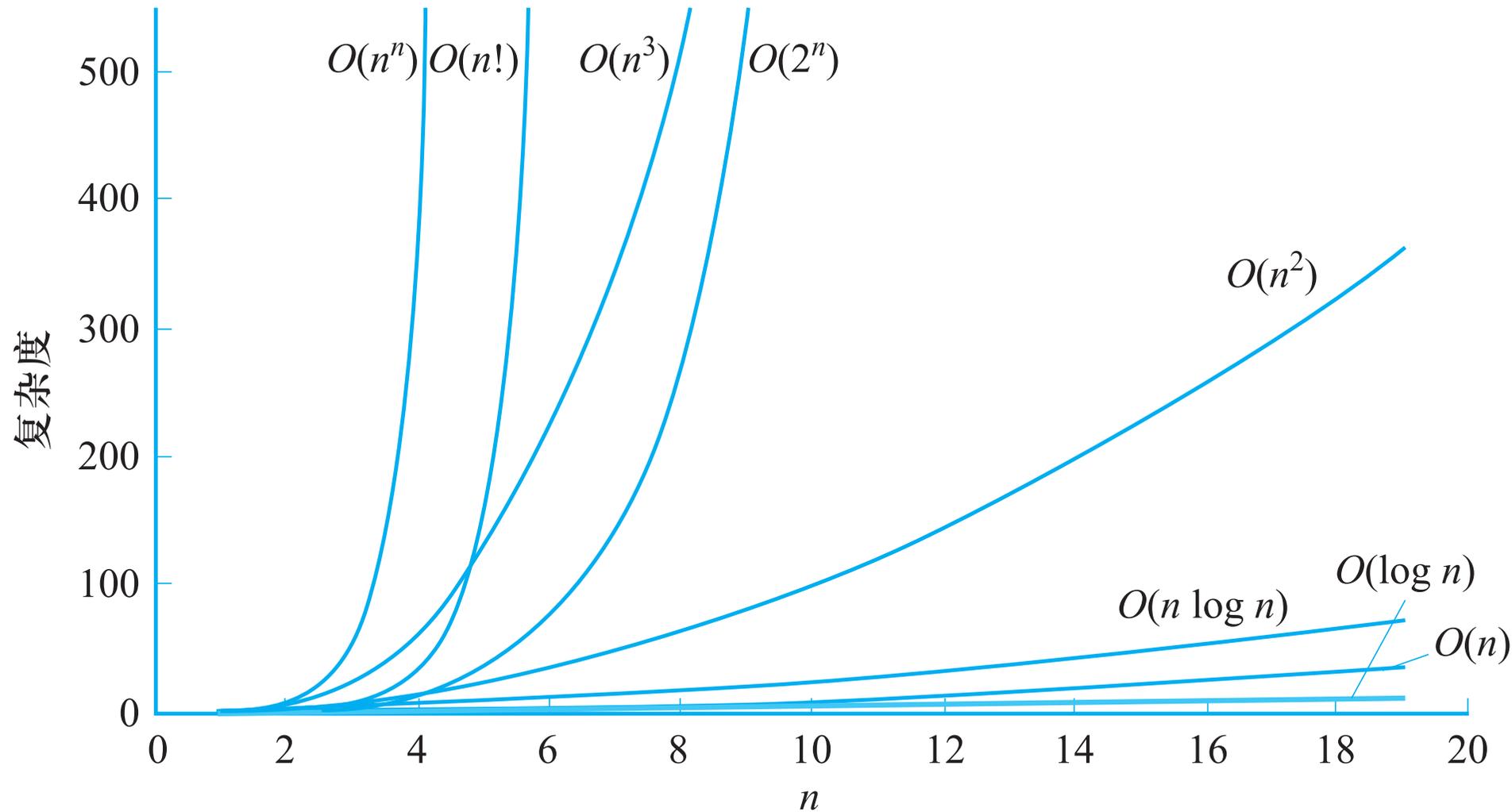
0.3.6 渐近复杂度的表示方式

定义 0.3 (大 O 表示法) 若存在一个常数 $c > 0$ 和整数 $n_0 > 0$, 使得对所有 $n \geq n_0$, 满足 $T(n) \leq cf(n)$, 则称 $T(n) = O(f(n))$ 。

例. $T(n) = 3n^2 + 100n$ 是 $O(n^2)$, 因为当 $n \geq 100$ 时, 设 $c = 4$, 对所有的 n , 都有 $T(n) \leq cn^2$ 。

大 O 表示法为待求解表达式 $T(n)$ 提供了一个**上界**, 描述其最多消耗 $f(n)$ 数量级的时间。

0.3.7 常见的复杂度



0.3.8 复杂度示例：选择排序算法

伪码 0.2: 选择排序

SelectionSort(A):

```
1   $n \leftarrow A.size$ 
2  for  $i \leftarrow 0$  to  $n - 2$ 
3       $k \leftarrow i$ 
4      for  $j \leftarrow i + 1$  to  $n - 1$ 
5          if  $A[j] > A[k]$ 
6               $k \leftarrow j$ 
7       $A[i] \leftrightarrow A[k]$ 
```

以上算法的时间复杂度是 $O(n^2)$ 。

0.4 课程信息

0.4.1 教学目标

- 掌握数据结构与算法的基本概念和基础知识
- 学习编写正确、高效的程序所必须掌握的数据结构与基本算法
- 提升综合运用理论知识，解决实际问题的基本技能

参考书：计算机领域本科教育教学改革试点工作（“101 计划”）系列教材《**数据结构**》，俞勇、张铭、陈越、韩文弢，高等教育出版社，2024

0.4.2 考核方式

1. **书面作业** (30%)：共 3 次，提交到网络学堂。
2. **上机作业** (30%)：共 3 次（不计热身题），提交到课程配套的在线评测系统。
3. **期末测验** (40%)：闭卷考试，于第 16 周课堂内进行。
4. **额外加分** (+5%)：奖励积极参加课堂互动、对课程建设有贡献、自主进行的与课程相关的探索等表现。

要求：

- 自主完成作业，鼓励讨论交流，**禁止抄袭**
- 鼓励使用人工智能工具辅助学习，提倡在理解的基础上使用

0.5 本讲小结

0.5.1 本讲小结

- 数据结构与算法的基本概念
- 数据的逻辑结构与存储结构
- 算法的定义与特性
- 复杂度的渐近表示法