

数据结构 01

2025 年 2 月 27 日

考纲内容:

- 绪论: 数据、数据元素、数据结构、数据类型、抽象数据类型的概念, 数据的逻辑结构和存储结构, 算法、算法描述和算法分析的概念。
- 线性表: 线性表的定义及其抽象数据类型描述, 顺序表的逻辑结构定义及其基本运算, 链表的逻辑结构及其基本操作。

1 绪论

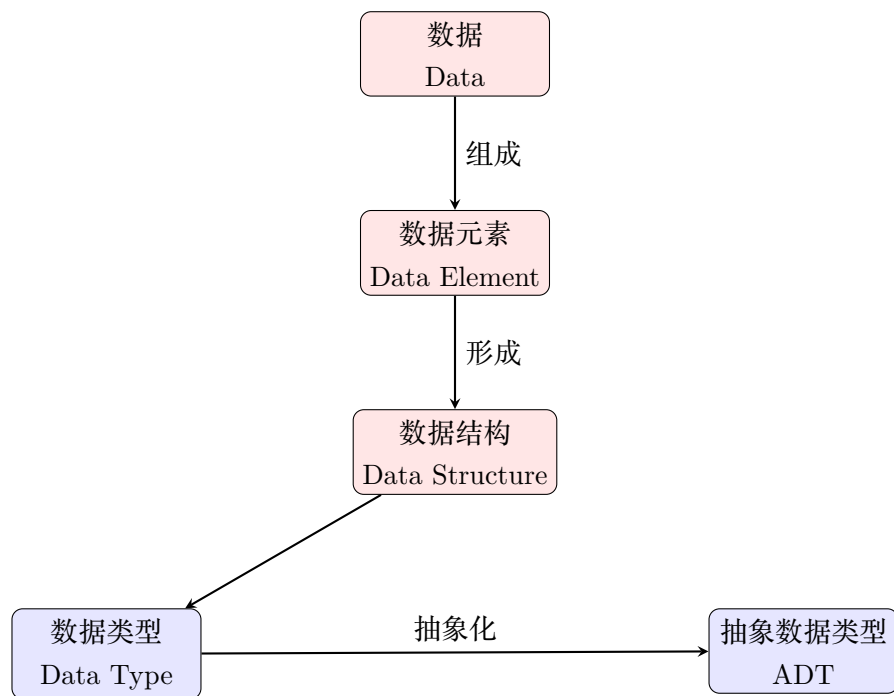


图 1: 数据结构基本概念关系图

1.1 基本概念

1.1.1 数据

数据 (Data) 是对客观事物的符号表示, 是信息的载体, 是计算机程序加工的对象。数据是计算机科学研究的基本对象, 也是计算机处理的对象。

1.1.2 数据元素

数据元素 (Data Element) 是数据的基本单位，是数据集合中的一个客体，也称为记录。数据元素通常由若干个数据项组成，每个数据项是数据元素的一个最小单位，也称为字段。

1.1.3 数据结构

数据结构 (Data Structure) 是指相互之间存在一种或多种特定关系的数据元素的集合。一个数据结构包括三个方面：

1. 数据的逻辑结构
2. 数据的存储结构
3. 数据的操作

1.1.4 数据类型

数据类型 (Data Type) 是指一组性质相同的值的集合及定义在此集合上的一组操作的总称。数据类型可分为基本数据类型（如整型、实型、字符型等）和结构化数据类型（如数组、记录、文件等）。

1.1.5 抽象数据类型

抽象数据类型 (Abstract Data Type, ADT) 是指一个数学模型以及定义在该模型上的一组操作。抽象数据类型把数据类型的逻辑特性与其实现细节分离，使用 ADT 来描述数据结构时只需关注其逻辑特性，而不必关心其具体实现。

1.2 数据的结构

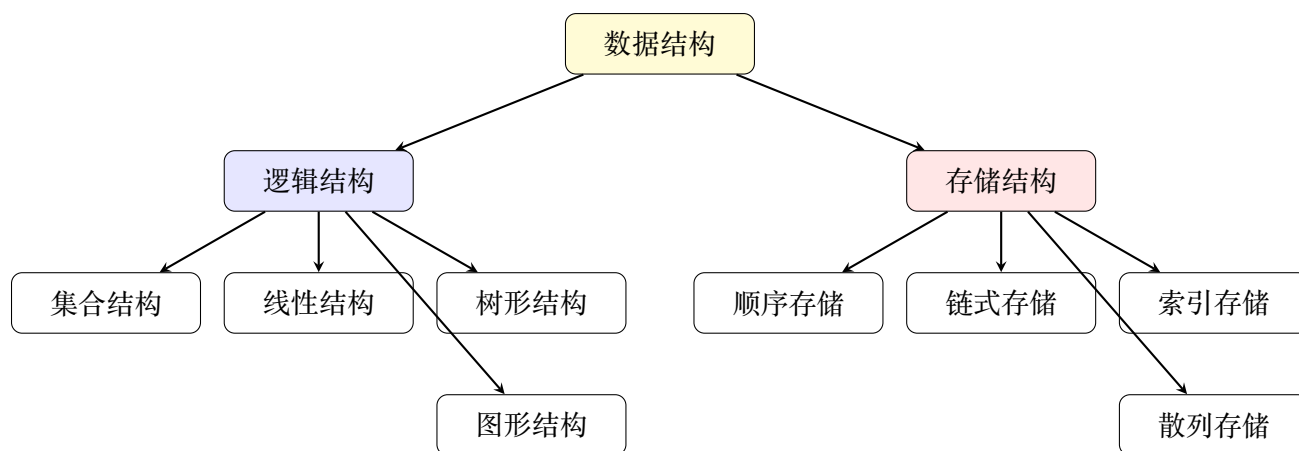


图 2: 数据结构的分类

1.2.1 数据的逻辑结构

逻辑结构 (Logical Structure) 是指数据元素之间的逻辑关系，与数据的存储无关。主要分为以下四类：

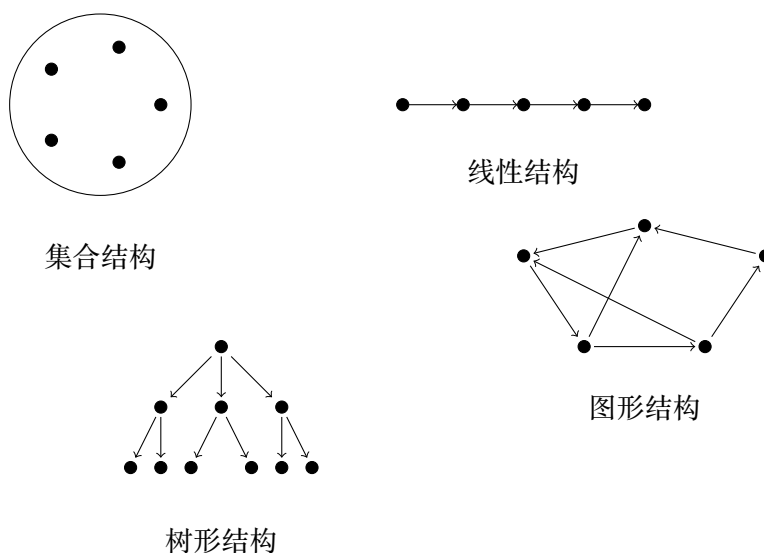


图 3: 数据的逻辑结构示意图

1. 集合结构：数据元素之间除了同属一个集合外，无其他关系
2. 线性结构：数据元素之间存在一对一的线性关系
3. 树形结构：数据元素之间存在一对多的层次关系
4. 图形结构：数据元素之间存在多对多的任意关系

1.2.2 数据的存储结构

存储结构 (Storage Structure) 是指数据结构在计算机中的表示，也称为物理结构。主要分为以下四类：

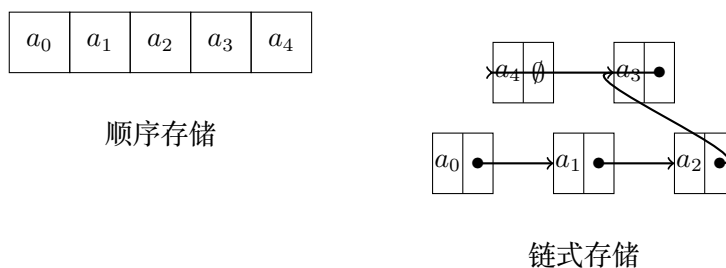


图 4: 顺序存储与链式存储结构示意图

1. 顺序存储：将数据元素存储在地址连续的存储单元中

2. 链式存储：将数据元素存储在任意的存储单元中，通过指针来表示元素之间的逻辑关系
3. 索引存储：在存储元素信息的同时，还建立附加的索引表
4. 散列存储：根据数据元素的关键字直接计算出该元素的存储地址

1.3 算法

1.3.1 算法的概念

算法 (Algorithm) 是对特定问题求解步骤的一种描述，是一组有穷的、确定的、可行的指令序列。算法具有五个基本特性：

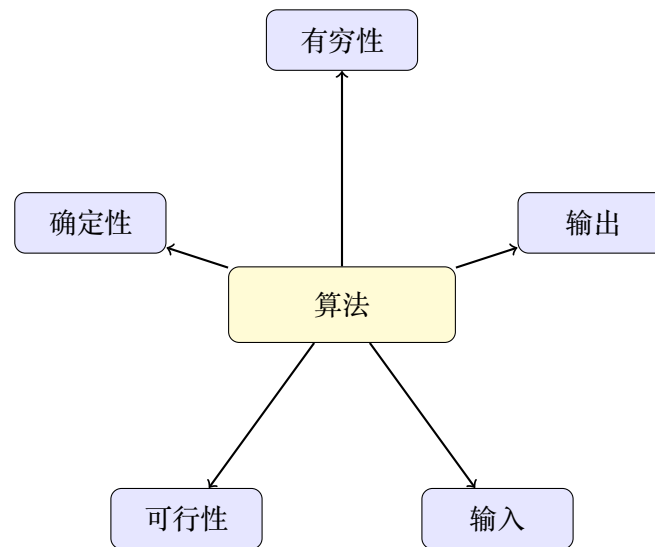


图 5: 算法的五个基本特性

1. 有穷性：算法必须在有限的步骤后终止
2. 确定性：每一步骤都有明确的定义
3. 可行性：每一步操作都是可以实现的
4. 输入：有零个或多个输入
5. 输出：有一个或多个输出

1.3.2 算法描述

算法描述是对算法进行表达的方式，常用的描述方法包括：

1. 自然语言描述
2. 流程图描述

3. 伪代码描述
4. 程序语言描述

1.3.3 算法分析

算法分析 (Algorithm Analysis) 是指对算法的执行效率和所需资源进行评估和预测。主要包括：

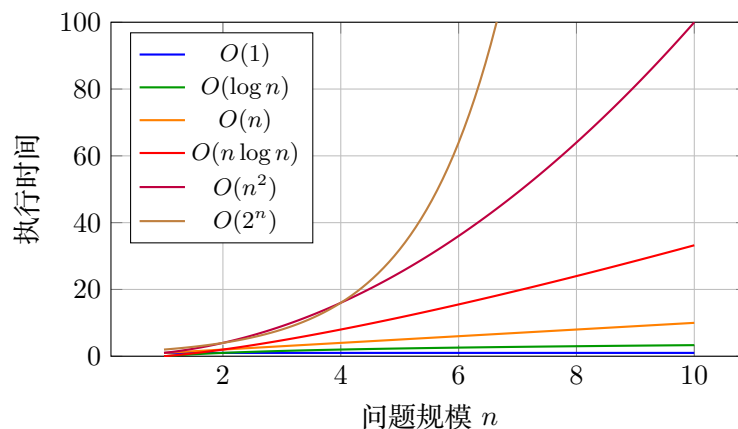


图 6: 常见时间复杂度增长率比较

1. 时间复杂度：评估算法执行所需的时间
2. 空间复杂度：评估算法执行所需的存储空间

时间复杂度通常用大 O 符号表示，常见的时间复杂度有： $O(1)$ 、 $O(\log n)$ 、 $O(n)$ 、 $O(n \log n)$ 、 $O(n^2)$ 、 $O(n^3)$ 、 $O(2^n)$ 等。

2 线性表

2.1 线性表的定义

线性表 (Linear List) 是具有相同数据类型的 n 个数据元素的有限序列。其中：

- 数据元素的个数 n 定义为表的长度
- 当 $n=0$ 时称为空表
- 表中任一元素所在位置后有且仅有一个直接后继元素
- 表中除第一个元素外，任一元素有且仅有一个直接前驱元素

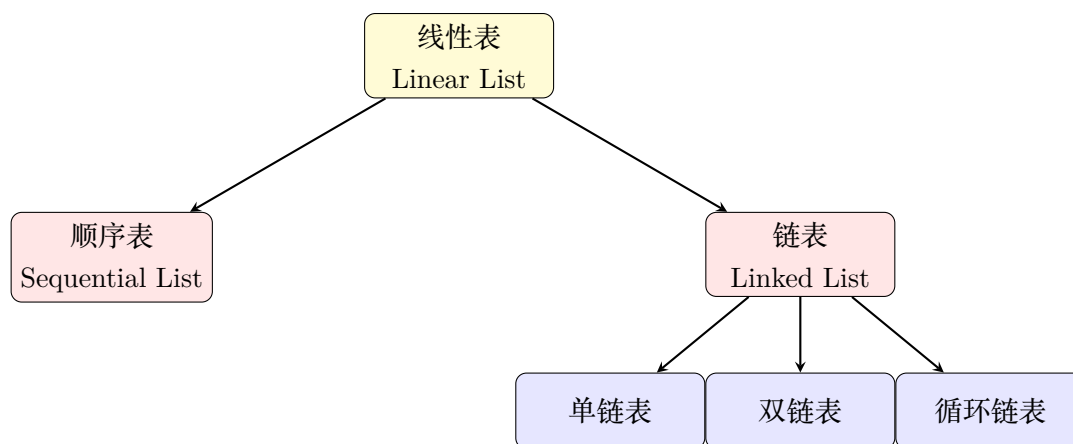


图 7: 线性表的分类及关系

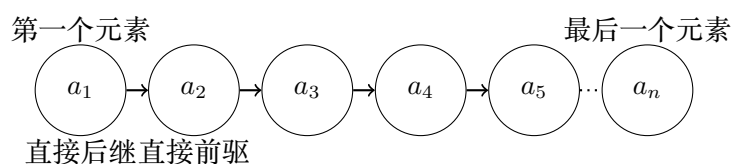


图 8: 线性表的逻辑结构示意图

2.2 线性表的抽象数据类型描述

线性表的 ADT 可以用以下方式表述：

- 数据对象：线性表是 n 个数据元素的有限序列
- 数据关系：除最后一个元素外，每个元素有且仅有一个直接后继；除第一个元素外，每个元素有且仅有一个直接前驱
- 基本操作：初始化、销毁、插入、删除、查找、更新等

2.3 顺序表

2.3.1 顺序表的定义

顺序表 (Sequential List) 是用一组地址连续的存储单元依次存储线性表的数据元素，从而使得逻辑上相邻的两个元素在物理位置上也相邻。

2.3.2 顺序表的特点

- 优点：随机访问（通过位置直接计算地址），存储密度高
- 缺点：插入删除需要移动元素，大小固定难以扩展

线性表 ADT

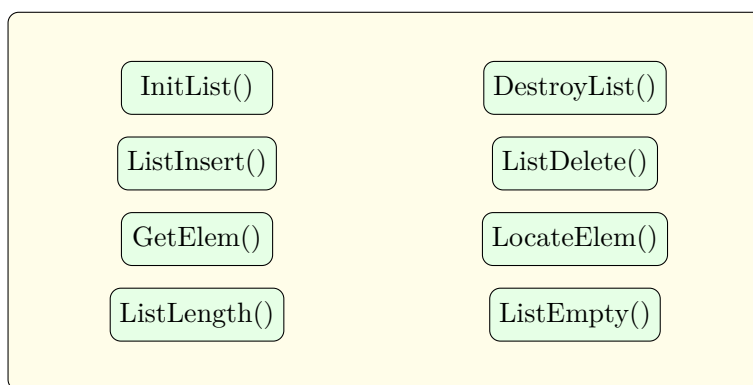


图 9: 线性表的抽象数据类型

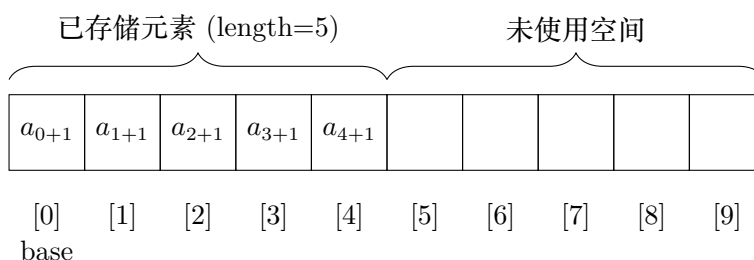


图 10: 顺序表的存储结构

2.3.3 顺序表的基本运算

顺序表的基本运算时间复杂度:

操作	最好情况	最坏情况
按值查找	$O(1)$	$O(n)$
按位查找	$O(1)$	$O(1)$
插入操作	$O(1)$	$O(n)$
删除操作	$O(1)$	$O(n)$

2.4 链表

2.4.1 链表的定义

链表 (Linked List) 是用一组任意的存储单元来依次存储线性表中的数据元素，元素的存储顺序不必和逻辑顺序一致，通过指针来指示元素的逻辑关系。

2.4.2 链表的特点

- 优点：插入删除不需要移动元素，只需修改指针，大小可动态调整
- 缺点：不支持随机访问，需要从头开始查找，存储密度低（需要额外的指针空间）

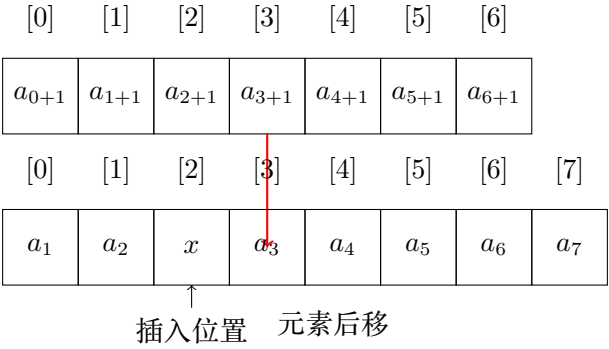


图 11: 顺序表的插入操作示意图

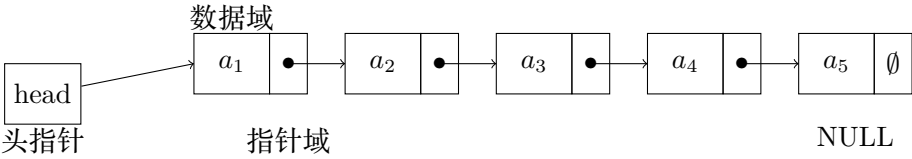


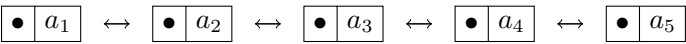
图 12: 单链表结构示意图

2.4.3 链表的分类

链表根据结构可分为：

1. 单链表：每个结点只有一个指针域，指向后继结点
2. 双链表：每个结点有两个指针域，分别指向前驱和后继结点
3. 循环链表：尾节点的指针指向头节点形成环状结构

双链表



循环链表

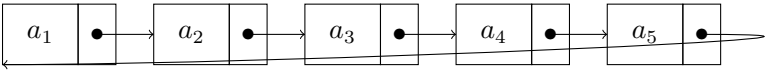


图 13: 双链表与循环链表示意图

2.4.4 链表的基本操作

链表的基本操作时间复杂度：

操作	顺序表	链表
按值查找	$O(n)$	$O(n)$
按位查找	$O(1)$	$O(n)$
插入操作	$O(n)$	$O(1)$
删除操作	$O(n)$	$O(1)$

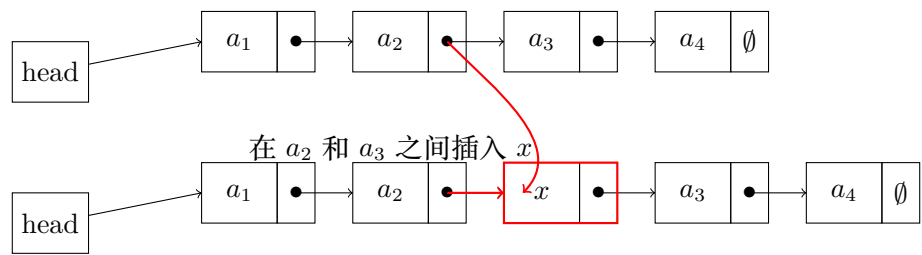


图 14: 链表的插入操作示意图