

第一章 概论

一. 重要概念

- 1.抽象数据结构
- 2.数据逻辑结构
- 3.数据存储结构
- 4.算法
- ★ 5. 算法分析(时间代价、空间代价)
- 6.数据结构的选择和评价

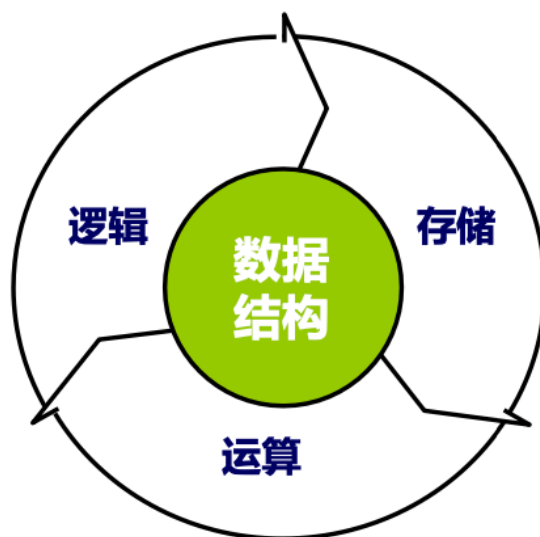
二. 方法

- 7.根据二元组画出图示逻辑结构(注意边的方向)
 - ★ 8. 根据要求设计数据结构
 - ★ 9. 算法的渐进分析方法
 - ★ 10.算法分析的大O表示法（不要求掌握大 Ω 、大 Θ 表示法）
-

- **结构: 实体 + 关系**

- **数据结构:**

- 按照逻辑关系组织起来的一批数据,
- 按一定的存储方法把它存储在计算机中
- 在这些数据上定义了一个运算的集合



1.抽象数据结构/抽象数据类型

概念

简称ADT

定义了一组运算的数学模型
与物理存储结构无关
使软件系统建立在数据之上（面向对象的模块化思想）

2.数据逻辑结构

概念

逻辑结构是从具体问题抽象出来的数学模型，反映了事物的组成结构及事物之间的逻辑关系

$$B = (K, R), K = a_0, a_1, \dots, a_{n-1}, R = r$$

线性结构

线性表（表、栈、队列、串...）

非线性结构

树(二叉树， Huffman树， 二叉检索树等)

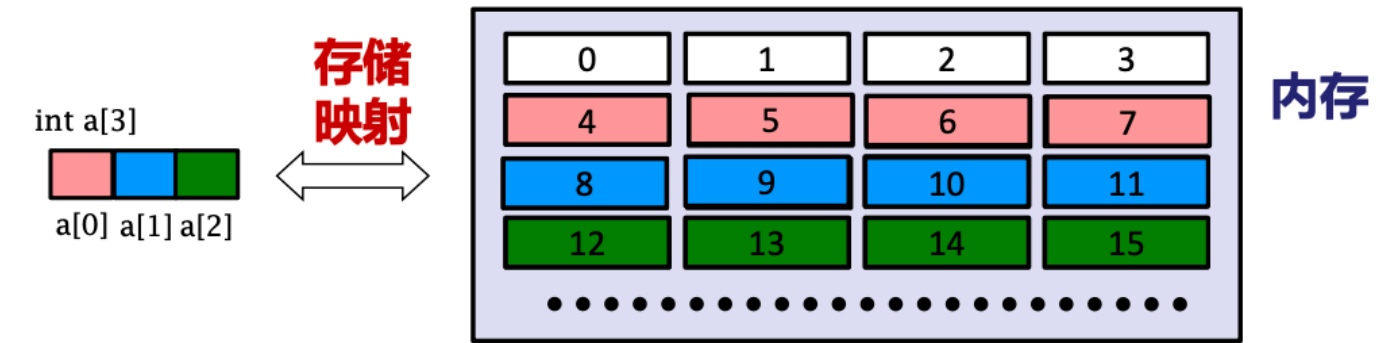
图(有向图， 无向图等)

图↗树↗二叉树↗线性表

3.数据存储结构

概念

对逻辑结构(K , r)，其中 $r \in R$ 。对结点集 K 建立一个从 K 到存储器 M 的单元的映射: $K \rightarrow M$ ，对于每一个结点 $j \in K$ 都对应一个**惟一**的连续存储区域 $c \in M$ 。



四类数据存储结构

顺序、链接、索引、散列

4.算法

概念

算法是求解问题的方法，是对特定问题求解过程的描述，是指令的有限序列

算法的特性

通用性 有效性 确定性 有穷性

基本算法的分类

穷举法 回溯、搜索 递归分治 贪心法 动态规划

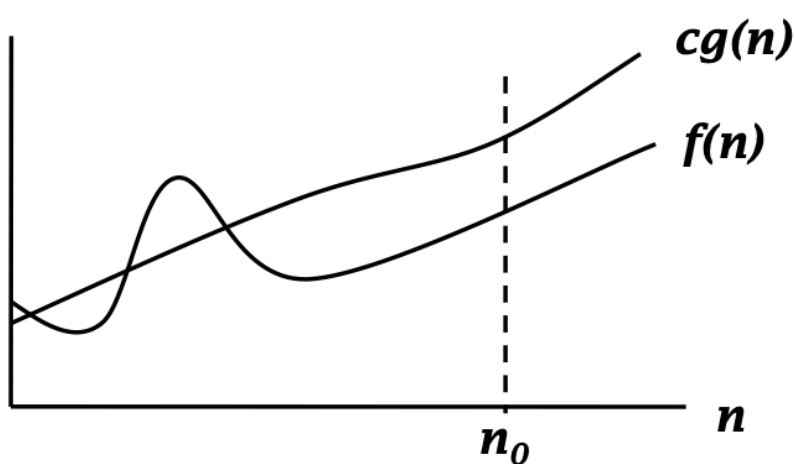
5.算法分析——大O表示法

概念

如果存在正数 c 和 n_0 ，使得对任意的 $n \leq n_0$ ，都有 $f(n) \geq cg(n)$ ，称 $f(n)$ 在集合 $O(g(n))$ 中，简称 $f(n)$ 是 $O(g(n))$ 的，或 $f(n) = O(g(n))$

含义

表示函数增长率上限——可能有多个上限，一般取上确界



**n 足够大
 $g(n)$ 是 $f(n)$ 的上界**

大O表示法的单位时间

简单布尔/算术运算

-变量或者数组链表元素简单读写操作

简单赋值

函数返回

简单I/O

-指函数的输入/输出参数传递

-不包括键盘、文件等I/O

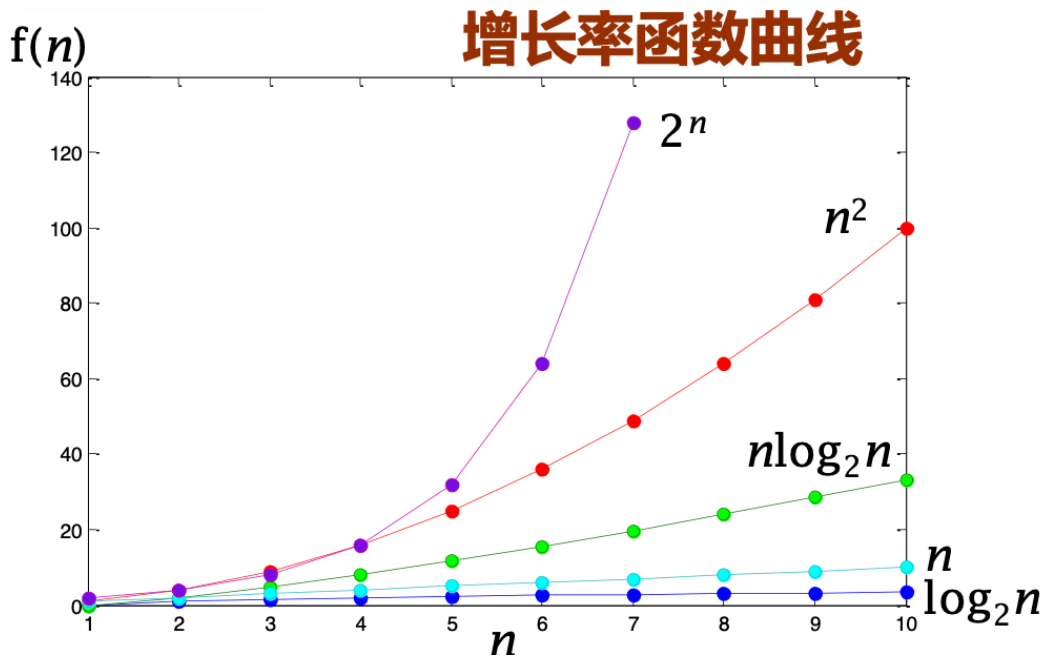
大O表示法的运算法则

加法法则——顺序结构，if结构，switch结构

$$O(f1(n) + f2(n)) = O(\max(f1(n), f2(n)))$$

乘法法则——for, while, do-while结构

$$O(f1(n) \cdot f2(n)) = O(f1(n) \cdot f2(n))$$



6. 数据结构的选择和评价

时空权衡

增大空间开销可能改善算法的时间开销

节省空间，往往需要增大运算时间

数据结构和算法的选择

仔细分析问题

-问题抽象、数据抽象

-数据结构的初步设计往往先于算法设计

注意数据结构的可拓展性

-考虑当输入数据的规模发生改变时，数据结构是否能够适应求解问题的演变和扩展