

## 引言

1 逻辑结构:线性结构->树->图

2 存储结构:顺序方法/链接方法/索引方法/散列方法

3 运算:增删改查/遍历排序

以上成为数据结构的三要素

### 1 逻辑结构

具体问题的数学抽象 反映事物的组成与逻辑关系

用一组数据(结点集合  $K$ ),数据间的二元关系(关系集合  $R$ ),表示为 $(K,R)$

$K$  由有限个结点组成,每个关系  $r \in R$  都是  $K * K$  上的二元关系

1 结点的类型:基本数据类型或复合数据类型

2 结构分类:逻辑结构分类,重点在关系集  $R$  上

集合结构

线性结构:前驱关系, $r$  是有向,满足全序性(全部结点可以比较前后),单索性(每个节点存在唯一前驱/后继结点)

树形结构:层次结构,每个节点唯一直接前驱,可有多于一个直接后继,根节点

图型结构:网络结构,对  $r$  没有任何约束

3 结点与结构:明确数据节点,主要关系  $r$ ;分析  $r$  时也分析数据类型;逻辑结构复杂,将其作为下一个层次,再分析下一层次结构

### 2 存储结构/物理结构

是逻辑结构的物理存储表示;计算机主存储器:空间相邻(非负整数编码,空间相邻的单元,字节作为基本单元)/随机访问(按地址随机访问,不同地址访问时间基本相同)

存储结构建立逻辑结构->物理结构的映射(结点->唯一的连续存储区域;逻辑关系->存储单元的地址顺序/指针指向关系)

基本的存储映射方法:

顺序方法:按地址相邻关系存储,逻辑关系用自然顺序关系表达,是紧凑存储结构(除有用数据外,无附加信息 存储密度:有用数据/结构整个存储空间)

链接方法:指针指向代表逻辑关系,数据字段=数据+指针字段,链索,增删容易,定位困难

索引方法:顺序存储推广,建立  $Y:Z \rightarrow$  存储地址  $D$  的函数,整数索引值->存储地址, $Y$  索引函数

散列方法:索引方法的拓展, $h:S \rightarrow$  非负整数  $Z$ (选择散列函数/建造散列表/解决碰撞的方法)

### 3 抽象数据类型

解决同类问题/是多种可能结构,实现的抽象/看做定义了操作的抽象类型/数学方法定义对象,运算集合,独立于计算机的表示方法/隐藏实现细节,内部数据结构

用三元组表示 $(D,S,P)$ 数据对象/数据关系/数据操作

### 1 算法性质

通用性 参数化输入求解,保证结果正确性

有效性 有限条指令完成功能,每步结果确定类型,可以预期

确定性 下一步执行步骤明确

有穷性 执行必须有限步内结束/不能含有死循环

### 2 算法分类

穷举法 对象有限/有穷举规则

回溯 试探/回溯

搜索 跳过无解分支,最优化问题通法

贪心法 最优子结构->最优解

递归分治 子结构不重复

动规 最优重复子结构/无后效性

### 3 算法分析

对时间存储空间定量分析/算法符合现实?/时空复杂性

不用绝对时间单位(环境/语言/规模)->复杂性与输入数据规模的关系

渐进分析->只考虑  $n$  幂次最高的项,常数项/低幂次项忽略不计

大  $O$  表示法:存在  $c, n_0$  使得  $f(n) \leq cg(n)$ , 对数函数无论底数为何,都具有相同的增长率

$$F_1(n) + f_2(n) = O(\max(f_1(n), f_2(n))) \quad F_1(n) * f_2(n) = O(f_1(n) * f_2(n))$$

$\Omega$  表示法:与大  $O$  相反,用来确定函数的下界

$\theta$  表示法:上下界相同时,可用  $\theta$  表示法,被同一函数控制

最好最坏和平均情况:受条件分支的影响,无法得出独立于输入数据的渐进估计

平均情况下的复杂性分析:考虑每种输入情况的概率,求期望

一般不注意算法的最好估计,只考虑最坏估计,一般情况下最坏情况和平均情况只有常数因子的区别,不会影响渐近分析的增长率函数估计

空间开销也可用类似的方法分析(静态存储结构/动态存储结构)

时空折中