一.数据结构

数据结构主要研究非数值计算程序问题中的操作对象以及他们之间的关系

二.基础概念

数据：程序的操作对象，用于描述客观事物

数据特点：可以输入到计算机

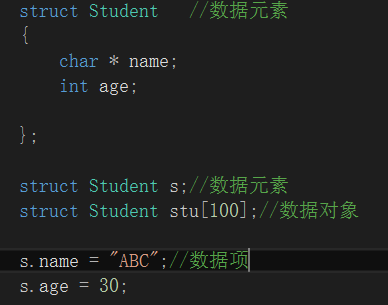
可以被计算机程序处理

数据是一个抽象的概念，将其进行分类得到程序设计语言中的类型，如：int ,float,char等

数据元素：组成数据的基本单位

数据项：一个数据元素由若干数据项组成

数据对象：性质相同的数据元素的集合



数据结构也即数据对象中数据元素之间的关系

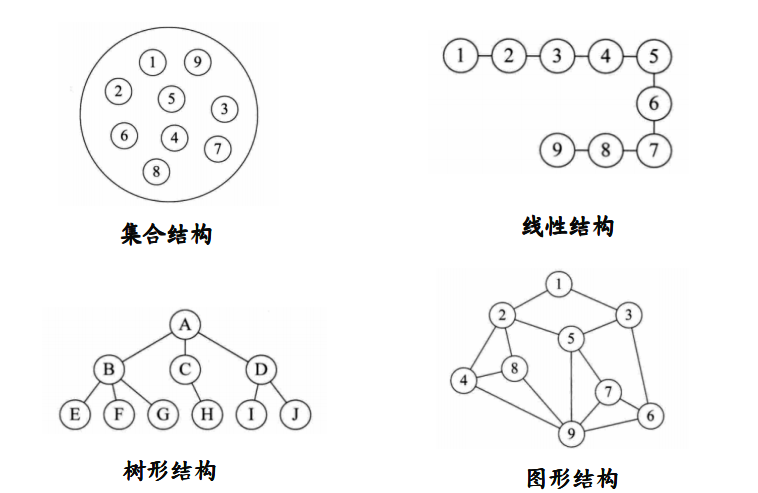
三.逻辑结构

集合结构：数据元素之间没有特别的关系，仅同属于相同集合

线性结构：数据元素之间是一对一的关系

树型结构：数据元素之间存在一对多的层次关系

图形结构：多对多



四.物理结构

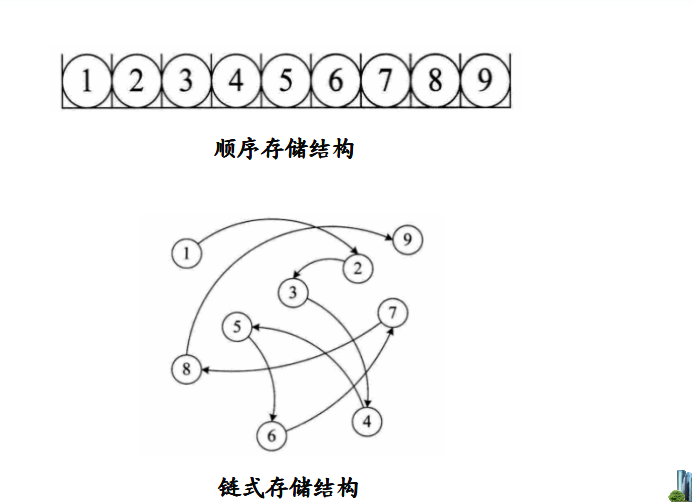
逻辑结构在计算机中的存储形式

主要包括：

顺序存储结构 将数据存储在地址连续的存储单元里

链式存储结构 将数据存储在任意的存储单元里，通过保

存地址的方式找到相关联的数据元素



五.算法

算法是特定问题求解步骤的描述，在计算机中表现为指令的有限序列。

特性：1.算法具有0个或多个输入

2.算法至少有1个或多个输出

3.算法在有限步骤之后会自动结束而不会无线循环

4.算法中的每一步都要确定的含义，不会出现二义性

5.算法的每一步都是可行的

六.算法效率的度量

1.影响算法的主要因素

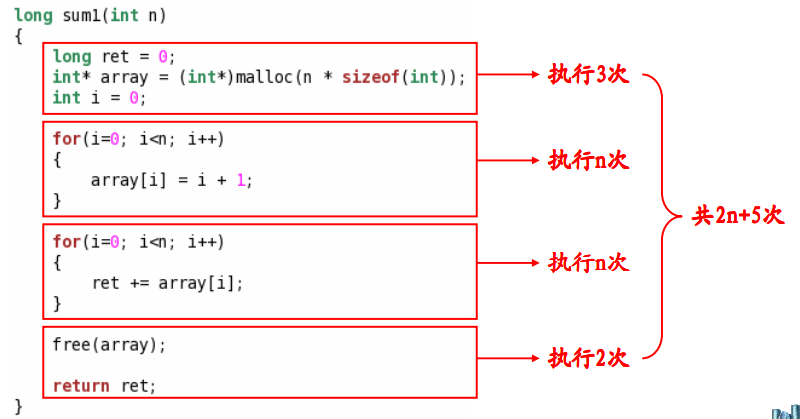
算法采用的策略和方法

问题的输入规模

编译器所产生的代码

计算机的执行速度

2.简单的效率估算



2.大O表示法

算法效率严重依赖于操作数量

在判读时首先关注操作数量的最高次项

操作数量的估算可以作为时间复杂度的估算

大O的运算

O(5) = O(1)

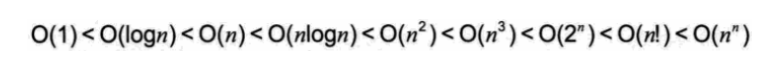
O(2n+1) = O(2n) = O(n)

O(n^2+n+1) = O(n^2)

O(3n^3+1) = O(3n^3) = O(n^3)

3.常用的复杂度类型





4.算法的空间复杂度

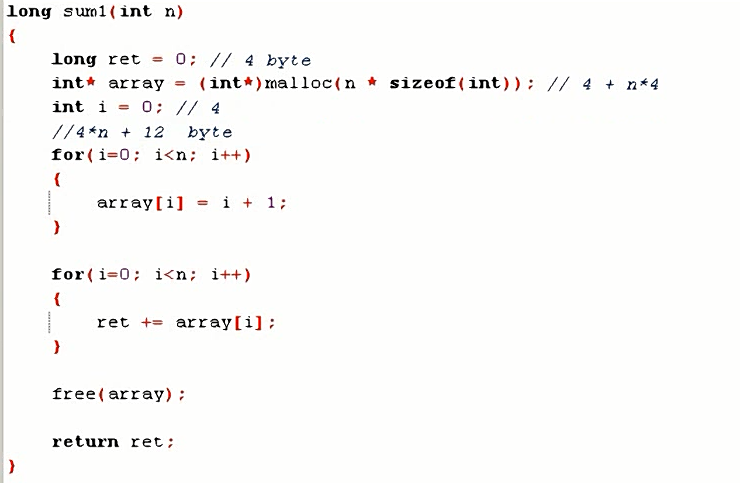
算法的空间复杂度通过计算算法的存储空间实现

S(n) = O(f(n))

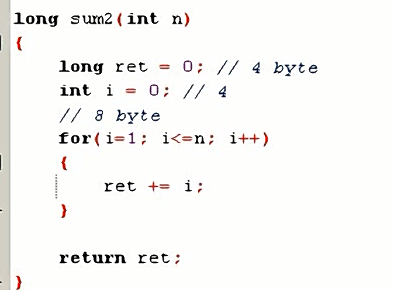
其中n为问题规模，f(n)为问题规模为n时占用的存储空间的函数

大O表示法适用于算法的空间复杂度

当算法执行时所需要的空间是常数时，空间复杂度为O(1)

如：

上面的算法占用了4n+12字节，所以O（4n+12） = O(n)



上面的占用了8个字节,所以O（8） = O（1）

5.空间与时间的策略

多数情况下，算法执行时所用的时间更令人关注

如有必要，可以通过增加空间复杂度来降低时间复杂度