

2.3线性表的链式表示(上)

背景

顺序表

优点：可以随时存取表中的任意一个元素
缺点：插入和删除操作需要移动大量元素

链式存储线性表

优点：不需要使用地址连续的存储单元，插入和删除操作不需要移动元素，而只需修改指针
缺点：失去顺序表可随机存取的优点

实现方式

带头结点

空表判断：L==NULL。代码书写不便

不带头结点

空表判断：L->next==NULL。写代码更方便

单链表的定义

线性表的链式存储又称单链表，它是指通过一组任意的存储单元来存储线性表中的数据元素

对每个链表结点，除存放元素自身的信息外，还需要存放一个指向其后继的指针

结构

data：数据域,存放数据元素

next：指针域,存放其后继结点的地址

优点

解决顺序表需要大量连续存储单元的缺点

缺点

单链表附加指针域，浪费存储空间
查找某个特定的结点时，需要从表头开始遍历，依次查找

从一个空表开始,生成新结点,并将读取到的数据存放到新结点的数据域中,然后将新结点插入到当前链表的表头,即头结点之后

采用头插法建立单链表

示意图

时间复杂度

每个结点插入的时间复杂度为O(1)
设单链表长为n,则总时间复杂度为O(n)

优点：算法实现简单

缺点：生成的链表中结点的次序和输入数据的顺序不一致 可以利用这个特点进行链表转置

该方法将新结点插入到当前链表的表尾,为此必须增加一个尾指针 r,使其始终指向当前链表的尾结点

采用尾插法建立单链表

示意图

按序号查找结点值

时间复杂度O(n)

按值查找表结点

时间复杂度O(n)

链表的本身特点原因，查找只能依次遍历查找

插入结点操作

示意图

实现代码段

p=GetElem(L,i-1);
s->next=p->next;
p->next=s;

若在给定的节点下进行插入，则时间复杂度为O(1)

本算法主要的时间开销在于查找第i-1个元素,时间复杂度为O(n)