



数据结构与算法（二）

张铭 主讲

采用教材：张铭，王腾蛟，赵海燕 编写
高等教育出版社，2008.6（“十一五”国家级规划教材）

<https://pkumooc.coursera.org/bdsalgo-001/>

第二章 线性表

- 2.1 线性表
- 2.2 顺序表
- 2.3 链表
- 2.4 顺序表和链表的比较



2.4 线性表实现方法的比较

- 顺序表的主要优点
 - 没有使用指针，不用花费额外开销
 - 线性表元素的读访问非常简洁便利
- 链表的主要优点
 - 无需事先了解线性表的长度
 - 允许线性表的长度动态变化
 - 能够适应经常插入删除内部元素的情况
- 总结
 - 顺序表是存储静态数据的不二选择
 - 链表是存储动态变化数据的良方



顺序表和链表的比较

- 顺序表

- 插入、删除运算时间代价 $O(n)$ ，查找则可常数时间完成
- 预先申请固定长度的连续空间
- 如果整个数组元素很满，则没有结构性存储开销

- 链表

- 插入、删除运算时间代价 $O(1)$ ，但找第 i 个元素运算时间代价 $O(n)$
- 存储利用指针，动态地按照需要为表中新的元素分配存储空间
- 每个元素都有结构性存储开销



顺序表和链表存储密度

n 表示线性表中当前元素的数目，

P 表示指针的存储单元大小（通常为 4 bytes）

E 表示数据元素的存储单元大小

D 表示可以在数组中存储的线性表元素的最大数目

- 空间需求
 - 顺序表的空间需求为 DE
 - 链表的空间需求为 $n(P + E)$
- n 的临界值，即 $n > DE / (P+E)$
 - n 越大，顺序表的空间效率就更高
 - 如果 $P = E$ ，则临界值为 $n = D / 2$



应用场合的选择

- 顺序表不适用的场合
 - 经常插入删除时，不宜使用顺序表
 - 线性表的最大长度也是一个重要因素
- 链表不适用的场合
 - 当读操作比插入删除操作频率大时，不应选择链表
 - 当指针的存储开销，和整个结点内容所占空间相比其比例较大时，应该慎重选择



顺序表和链表的选择

- 顺序表
 - 结点总数目大概可以估计
 - 线性表中结点比较稳定（插入删除少）
 - $n > DE / (P + E)$
- 链表
 - 结点数目无法预知
 - 线性表中结点动态变化（插入删除多）
 - $n < DE / (P + E)$

思考

- 顺序表和链表的选择？
 - 结点变化的动态性
 - 存储密度

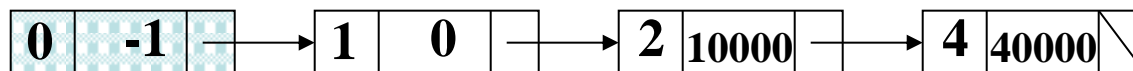
思考：一元多项式的表达

- 一元多项式： $P_n(x) = p_0 + p_1x + p_2x^2 + \dots + p_n x^n$
- 线性表表示： $P = (p_0, p_1, p_2, \dots, p_n)$
- 顺序表表示：只存系数（第 i 个元素存 x^i 的系数）



数据稀疏的情况： $p(x) = 1 + 2x^{10000} + 4x^{40000}$

- 链表表示： 结点结构





数据结构与算法

谢谢聆听

国家精品课“数据结构与算法”

<http://www.jpk.pku.edu.cn/pkujpk/course/sjjg/>

张铭，王腾蛟，赵海燕

高等教育出版社，2008. 6。“十一五”国家级规划教材