



日期: /

顺序表



考虑平均的情况:

假设在第 i 个元素之前插入的概率为 p_i ,
则在长度为 n 的线性表中插入一个元素所需移动元素次数的期望值为:

$$E_{is} = \sum_{i=1}^{n+1} p_i (n - i + 1)$$

若假定在线性表中任何一个位置上插入的概率都是相等的, 则移动元素的期望值为

$$E_{is} = \frac{1}{n+1} \sum_{i=1}^{n+1} (n - i + 1) = \frac{n}{2}$$



考虑平均的情况:

假设删除第 i 个元素的概率为 q_i ,
则在长度为 n 的线性表中删除一个元素所需移动元素次数的期望值为:

$$E_{dl} = \sum_{i=1}^n q_i (n - i)$$

若假定在线性表中任何一个位置上删除的概率都是相等的, 则移动元素的期望值为

$$E_{dl} = \frac{1}{n} \sum_{i=1}^n (n - i) = \frac{n-1}{2}$$



顺序表性能分析

在顺序表中进行(顺序)查找查找成功的平均查找长度为:

设表中每个元素的查找概率相等 $p_i = \frac{1}{n}$

$$\text{则 } ASL = \sum_{i=1}^n p_i c_i = \frac{1}{n} \sum_{i=1}^n i = \frac{1}{n} \cdot \frac{n(n+1)}{2} = \frac{n+1}{2}$$

❑ 查找不成功的平均查找长度为 $n+1$ 。

❑ 特点

- ❖ 优点: 算法简单且适应面广。
- ❖ 缺点: 其平均查找长度较大。

折半查找的特点

- 优点: 查找速度快; 但表必须有序。
- 缺点: 频繁插入和删除不方便。

- ❑ 折半查找适于表中元素很少变化且查找频繁的情况;
- ❑ 顺序表查找适于查找少而表中元素频繁变化的情况。

日期:

/

对于直接插入排序:

最好的情况 (关键字在记录序列中顺序有序):

“比较” 的次数:

“移动” 的次数:

$$\sum_{i=2}^n 1 = n - 1$$

0

最坏的情况 (关键字在记录序列中逆序有序):

“比较” 的次数:

“移动” 的次数:

$$\sum_{i=2}^n (i) = \frac{(n+2)(n-1)}{2}$$

$$\sum_{i=2}^n (i+1) = \frac{(n+4)(n-1)}{2}$$