



数据结构与算法（八）

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）

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

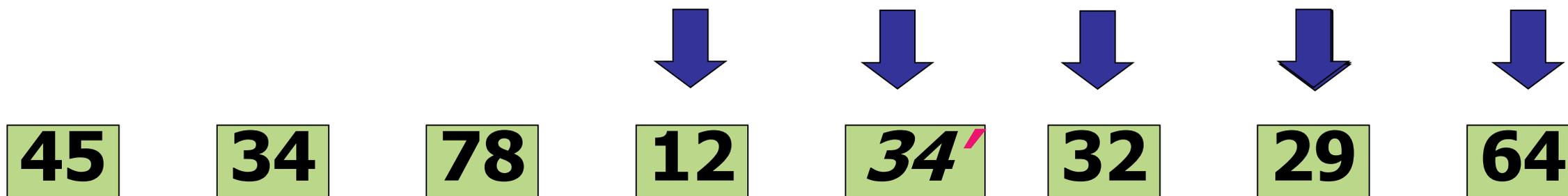
大纲

- 8.1 排序问题的基本概念
- 8.2 插入排序 (Shell 排序)
- **8.3 选择排序 (堆排序)**
- 8.4 交换排序
 - 8.4.1 冒泡排序
 - 8.4.2 快速排序
- 8.5 归并排序
- 8.6 分配排序和索引排序
- 8.7 排序算法的时间代价
- 内排序知识点总结

8.3 选择排序

- 8.3.1 直接选择排序
 - 依次选出剩下的未排序记录中的最小记录
- 8.3.2 堆排序
 - 堆排序：基于最大堆来实现

直接选择排序动画



直接选择排序

```
template <class Record>
void SelectSort(Record Array[], int n) {
// 依次选出第i小的记录，即剩余记录中最小的那个
    for (int i=0; i<n-1; i++) {
        // 首先假设记录i就是最小的
        int Smallest = i;
        // 开始向后扫描所有剩余记录
        for (int j=i+1; j<n; j++)
            // 如果发现更小的记录，记录它的位置
            if (Array[j] < Array[Smallest])
                Smallest = j;
        // 将第i小的记录放在数组中第i个位置
        swap(Array, i, Smallest);
    }
}
```



直接选择排序性能分析

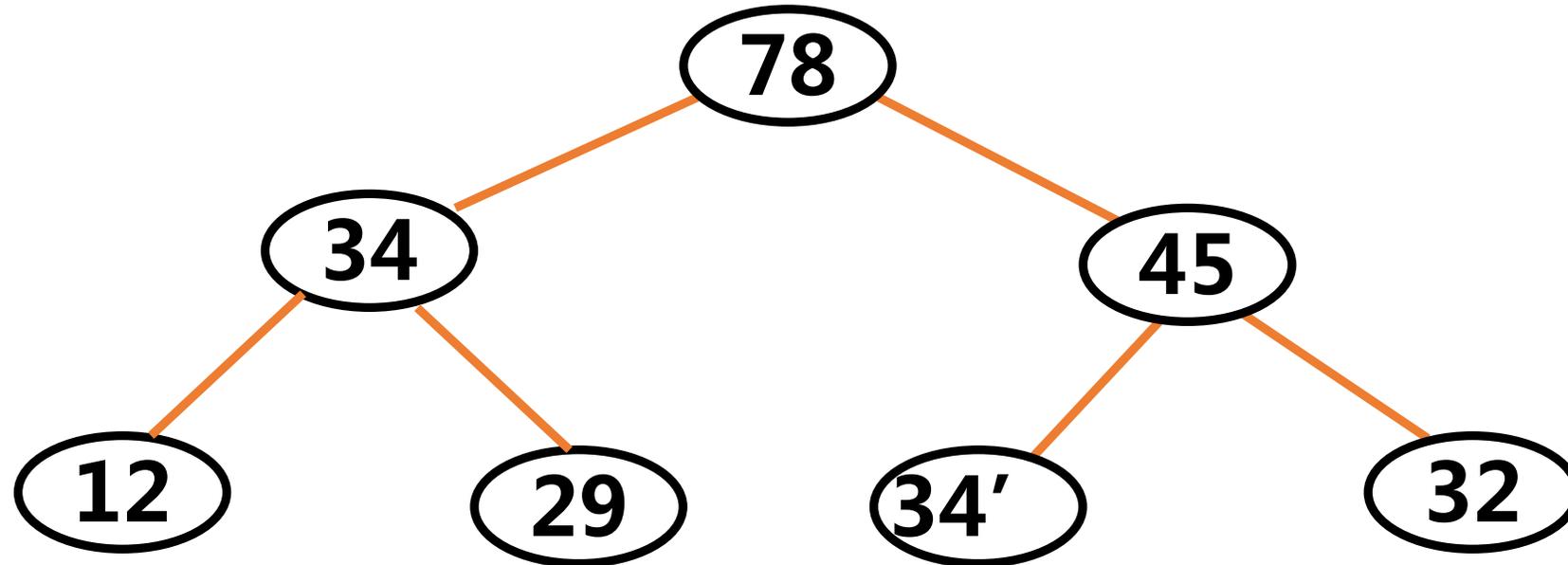
- 不稳定
- 空间代价： $\Theta(1)$
- 时间代价
 - 比较次数： $\Theta(n^2)$
 - 交换次数： $n-1$
 - 总时间代价： $\Theta(n^2)$

8.3.2 堆排序

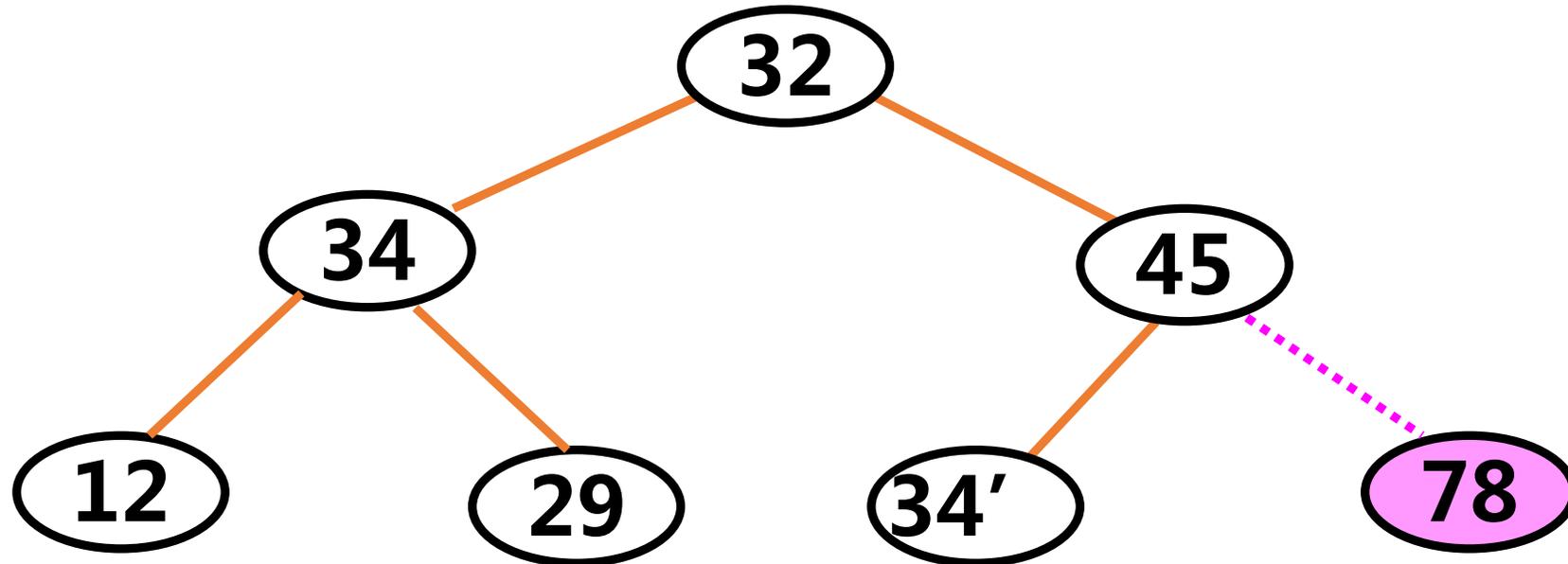
- 选择类内排序
 - 直接选择排序：直接从剩余记录中线性查找最大记录
 - 堆排序：基于最大堆来实现，效率更高
- 选择类外排序
 - 置换选择排序
 - 赢者树、败方树



最大堆排序过程示意图



最大堆排序过程示意图



堆排序算法

```
template <class Record>
void sort(Record Array[], int n){
    int i;
    // 建堆
    MaxHeap<Record> max_heap
        = MaxHeap<Record>(Array,n,n);
    // 算法操作n-1次，最小元素不需要出堆
    for (i = 0; i < n-1; i++)
        // 依次找出剩余记录中的最大记录，即堆顶
        max_heap.RemoveMax();
}
```

算法分析

- 建堆： $\Theta(n)$
- 删除堆顶： $\Theta(\log n)$
- 一次建堆， n 次删除堆顶
- 总时间代价为 $\Theta(n \log n)$
- 空间代价为 $\Theta(1)$

思考

- 直接选择排序为什么不稳定？怎么修改一下让它变稳定
- 改写堆排序算法，发现逆序对直接交换



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