Technology Tarena 达内科技

数据结构与算法

DATASTRUCTURE DAY04

内容

上午	09:00 ~ 09:30	作业讲解和回顾
	09:30 ~ 10:20	查找算法
	10:30 ~ 11:20	
	11:30 ~ 12:20	排序算法
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	
	16:00 ~ 16:50	
	17:00 ~ 17:30	总结和答疑

查找算法

Technology Tarena 达内科技

线性查找

线性查找

- 算法描述
  1. 从头开始，依次将每一个元素与查找目标进行比较
  2. 或者找到目标，或者找不到目标
- 伪码实现
  - **for** each item in the list
    if that item has the desired value
    stop the search and **return** the item's location
    return *not found*
- 总体评价
  - 平均时间复杂度：O(N)
  - 对数据的有序性没有要求

知识讲解

线性查找

【参见：TTS COOKBOOK 】

课堂练习

- 线性查找

Technology Tarena 达内科技

二分查找

二分查找

- 算法描述
  1. 假设表中的元素按升序排列
  2. 若中间元素与查找目标相等，则查找成功，否则利用中间元素将表划分成前后两个子表
  3. 若查找目标小于中间元素，则在前子表中查找，否则在后子表中查找
  4. 重复以上过程，直到查找成功，或者因子表不存在而宣告失败

知识讲解

二分查找（续1）

- 伪码实现
  - 基于递归的二分查找

```
function binarySearch (a, value, left, right)
    if right < left
        return not found
    mid := floor ((right - left) / 2) + left
    if a[mid] = value
        return mid
    if value < a[mid]
        return binarySearch (a, value, left, mid - 1)
    else
        return binarySearch (a, value, mid + 1, right)
```

知识讲解

二分查找（续2）

- 伪码实现
  - 基于循环的二分查找

```
function binarySearch (a, value, left, right)
    while left ≤ right
        mid := floor ((right - left) / 2) + left
        if a[mid] = value
            return mid
        if value < a[mid]
            right := mid - 1
        else
            left  := mid + 1
    return not found
```

知识讲解

二分查找（续3）

- 总体评价
  - 平均时间复杂度：O(logN)
  - 数据必须有序

知识讲解

二分查找

【参见：TTS COOKBOOK 】

课堂练习

- 二分查找

排序算法

Technology Tarena 达内科技

冒泡排序

冒泡排序

- 算法描述
  1. 相邻元素两两比较，前者大于后者，彼此交换
  2. 从第一对到最后一对，最大的元素沉降到最后
  3. 针对未排序部分，重复以上步骤，沉降次大值
  4. 每次扫描越来越少的元素，直至不再发生交换

知识讲解

第一趟

30 20 50 40 10

20 30 50 40 10

20 30 50 40 10

20 30 40 50 10

20 30 40 10 50

第二趟

20 30 40 10 50

20 30 40 10 50

20 30 40 10 50

20 30 10 40 50

第三趟

20 30 10 40 50

20 30 10 40 50

20 10 30 40 50

第四趟

20 10 30 40 50

10 20 30 40 50

结果

10 20 30 40 50

冒泡排序（续1）

- 伪码实现
  - **procedure** bubbleSort (A : list of sortable items)

```
    n = length (A)
    repeat
        swapped = false
        for i = 1 to n-1 inclusive do
            if A[i-1] > A[i] then
                swap (A[i-1], A[i])
                swapped = true
            end if
        end for
        n = n - 1
    until not swapped
end procedure
```

知识讲解

冒泡排序（续2）

- 总体评价
  - 平均时间复杂度：$O(N^2)$
  - 稳定排序
  - 对数据的有序性非常敏感

知识讲解

冒泡排序

【参见：TTS COOKBOOK 】

课堂练习

- 冒泡排序

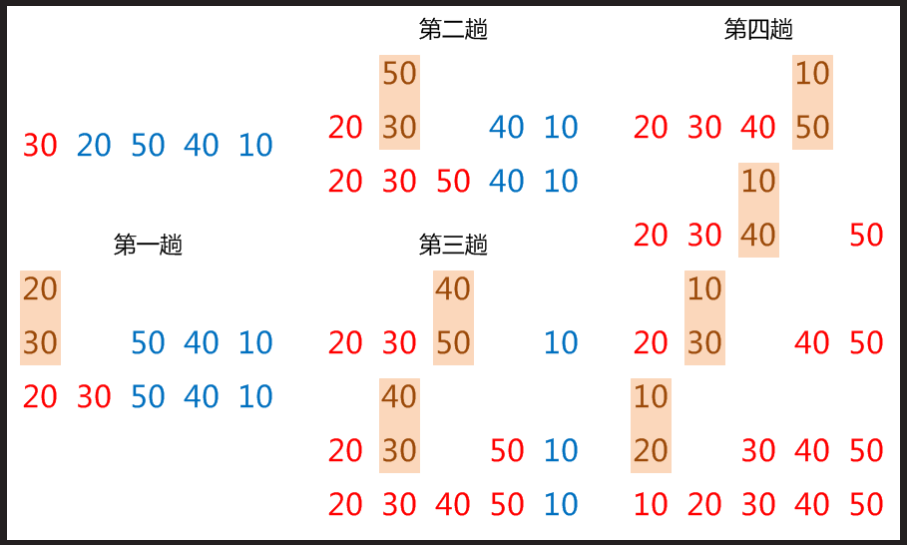
Technology Tarena 达内科技

插入排序

插入排序

- 算法描述
  1. 首元素自然有序
  2. 取出下一个元素，对已排序序列，从后向前扫描
  3. 大于被取出元素者后移
  4. 小于等于被取出元素者，将被取出元素插入其后
  5. 重复步骤2，直至处理完所有元素

知识讲解



插入排序（续1）

- 伪码实现
  - **for** i = 1 **to** length (A)
    x = A[i]
    j = i
    while j > 0 and A[j-1] > x
      A[j] = A[j-1]
      j = j - 1
    A[j] = x
- 总体评价
  - 平均时间复杂度：$O(N^2)$
  - 稳定排序
  - 对数据的有序性非常敏感
  - 不交换只移动，优于冒泡排序

知识讲解

插入排序

【参见：TTS COOKBOOK 】

课堂练习

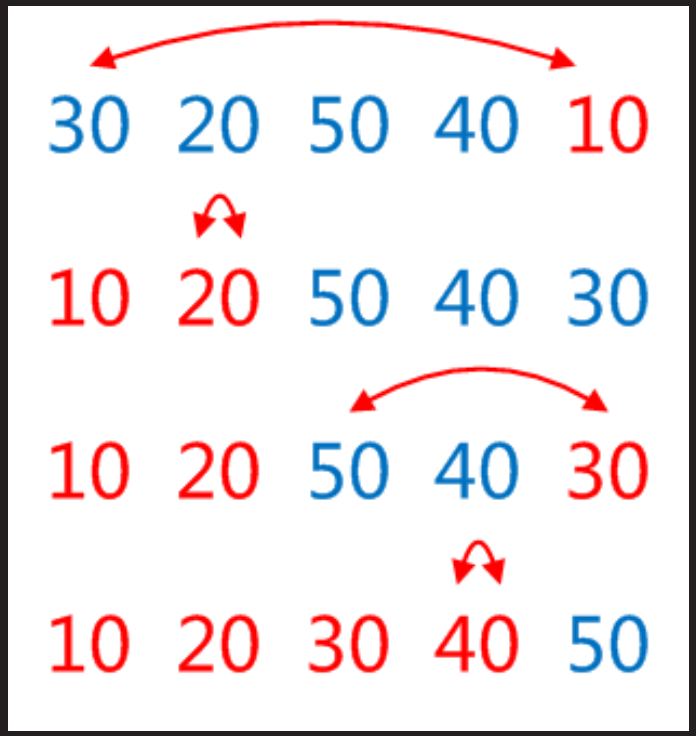
- 插入排序

Technology Tarena 达内科技

选择排序

选择排序

- 算法描述
  1. 在整个序列中寻找最小元素，与首元素交换
  2. 在剩余序列中寻找最小元素，与次元素交换
  3. 以此类推，直到剩余序列中仅包含一个元素



知识讲解

选择排序（续1）

- 伪码实现
  - **function** selectionSort (list[1..n])
    for i **from** 1 **to** n - 1
    minIndex = i
    for j **from** i + 1 **to** n
    if list[j] < list[minIndex]
    minIndex = j
    swap list[i] and list[minIndex]
- 总体评价
  - 平均时间复杂度：$O(N^2)$
  - 非稳定排序
  - 对数据的有序性不敏感
  - 交换次数少，优于冒泡排序

知识讲解

Technology Tarena 达内科技

选择排序

【参见：TTS COOKBOOK 】

课堂练习

- 选择排序

Technology Tarena 达内科技

快速排序

快速排序

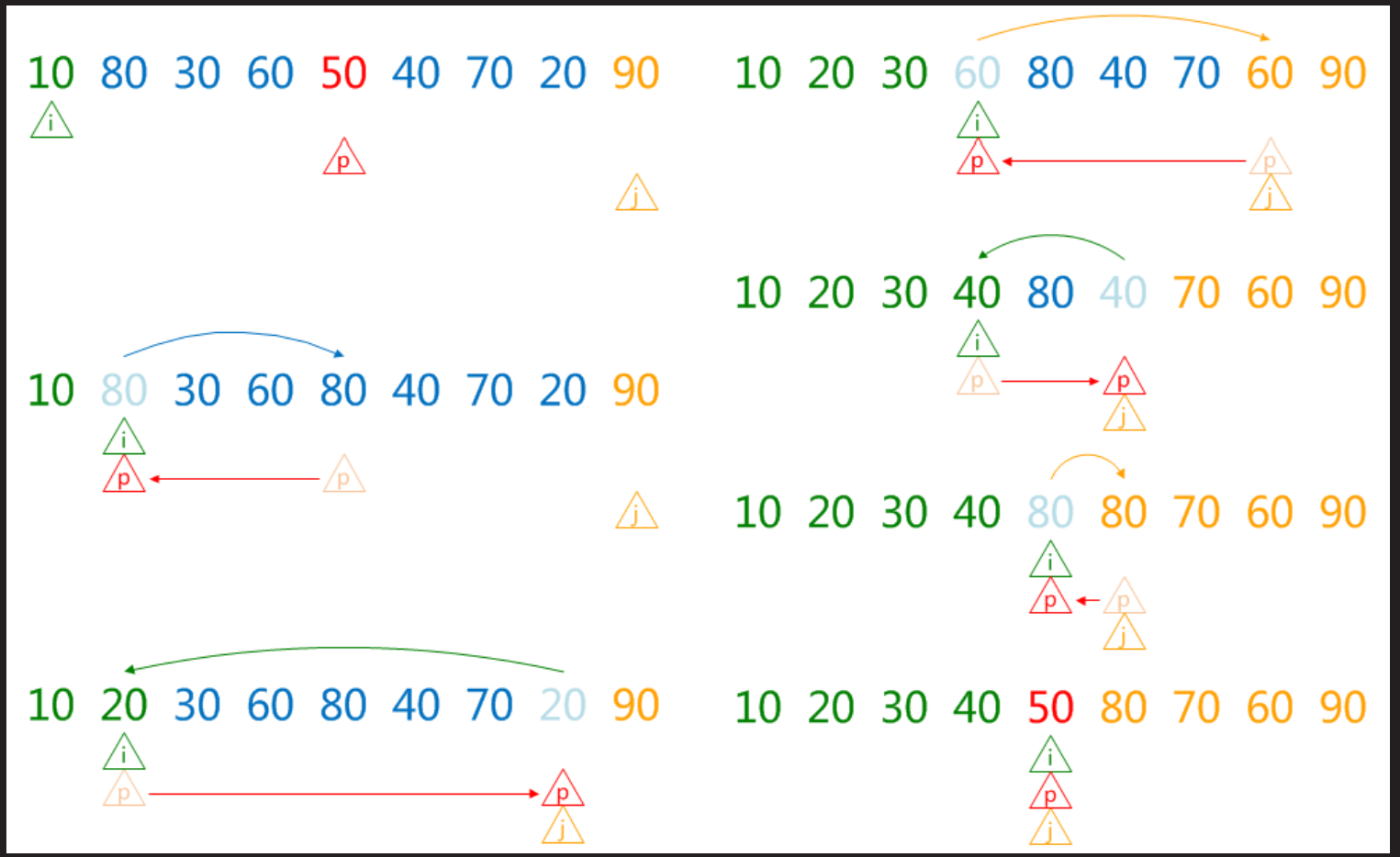
- 算法描述
  1. 从待排序序列中任意挑选一个元素，作为基准
  2. 将所有小于基准的元素放在基准之前，大于基准的元素放在基准之后，等于基准的元素放在基准之前或之后，这个过程称为分组
  3. 以递归的方式，分别对基准之前和基准之后的分组继续进行分组，直到每个分组内的元素个数不多于1个为止

知识讲解

10	80	30	60	50	40	70	20	90
10	20	30	40	50	80	70	60	90
10	20	30	40	50	60	70	80	90
10	20	30	40	50	60	70	80	90

快速排序（续1）

- 就地分组
  - 在不额外分配内存空间的前提下，完成分组过程



知识讲解

快速排序（续2）

- 伪码实现
  - 基于分组的排序

```
quickSort (A, i, k):
    if i < k:
        p := partition (A, i, k)
        quickSort (A, i, p - 1)
        quickSort (A, p + 1, k)
```

知识讲解

快速排序（续3）

- 伪码实现
  - 就地分组

```
partition (array, left, right)
    pivotIndex := choosePivot (array, left, right)
    pivotValue := array[pivotIndex]
    i = left
    j = right
    while i < j
        while i < pivotIndex and array[i] ≤ pivotValue
            i = i + 1
        if i < pivotIndex
            array[pivotIndex] = array[i]
            pivotIndex = i
        while j > pivotIndex and array[j] ≥ pivotVAlue
            j = j - 1
        if j > pivotIndex
            array[pivotIndex] = array[j]
            pivotIndex = j
    array[pivotIndex] = pivotValue
    return pivotIndex
```

知识讲解

快速排序（续4）

- 总体评价
  - 平均时间复杂度：O(NlogN)
  - 非稳定排序
  - 若每次都能均匀分组，则排序速度最快

知识讲解

快速排序

【参见：TTS COOKBOOK 】

课堂练习

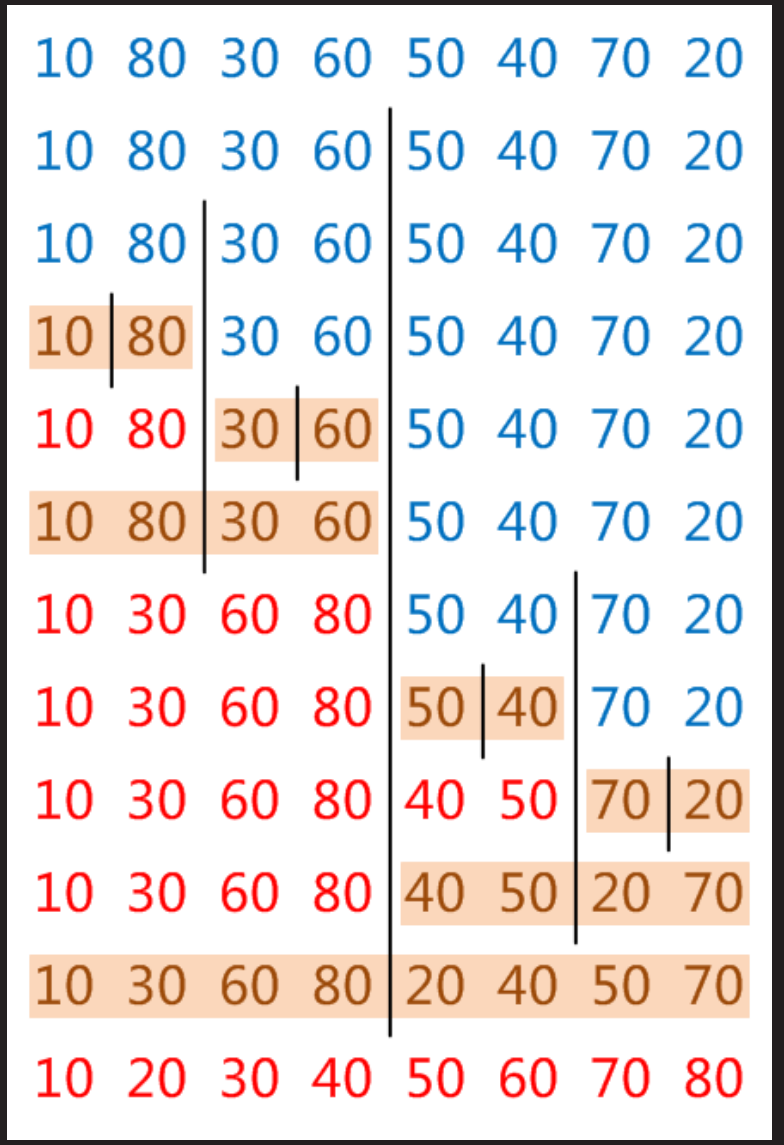
- 快速排序

Technology Tarena 达内科技

归并排序

归并排序

- 算法描述
  1. 将待排序序列从中间划分为两个相等的子序列
  2. 以递归的方式分别对两个子序列进行排序
  3. 将两个有序的子序列合并成完整序列



知识讲解

归并排序（续1）

- 有序合并
  1. 分配合并序列，其大小为两个有序序列大小之和
  2. 设定两个指针，分别指向两个有序序列的首元素
  3. 比较指针目标，较小者进入合并序列，指针后移
  4. 重复步骤3，直到某一指针到达序列末尾
  5. 将另一序列的剩余元素直接复制到合并序列末尾

知识讲解

归并排序（续2）

- 有序合并

知识讲解

归并排序（续3）

- 伪码实现
  - 基于合并的排序

```
function mergeSort (list m)
    if length (m) <= 1
        return m
    var list left, right
    var integer middle = length (m) / 2
    for each x in m before middle
        add x to left
    for each x in m after or equal middle
        add x to right
    left = mergeSort (left)
    right = mergeSort (right)
    return merge (left, right)
```

归并排序（续4）

- 伪码实现
  - 有序合并

```
function merge (left, right)
    var list result
    while length (left) > 0 or length (right) > 0
        if length (left) > 0 and length (right) > 0
            if first (left) <= first (right)
                append first (left) to result
                left = rest (left)
            else
                append first (right) to result
                right = rest (right)
        else if length (left) > 0
            append first (left) to result
            left = rest (left)
        else if length (right) > 0
            append first (right) to result
            right = rest (right)
    end while
    return result
```

知识讲解

归并排序（续5）

- 总体评价
  - 平均时间复杂度：O(2NlogN)
  - 稳定排序
  - 对数据的有序性不敏感
  - 非就地排序，需要辅助空间，不适合处理海量数据

知识讲解

归并排序

【参见：TTS COOKBOOK 】

课堂练习

- 归并排序

Technology Tarena 达内科技

总结和答疑