

算法重点

- **题型**

- 判断题*10
 - 可能涉及非重点
- 简答*2
 - 变动大
 - NP
 - 问题间的包含关系
 - 定义, 如何证明
- 计算*4
 - 计算量大
 - Floyd-Warshall
- 算法设计*3
 - 要给算法思路, 给出中间核心部分代码就行
 - 动态规划
 - 贪心/查找、排序

- **重点**

- O符号

O符号

讨论 $O(g(n))$ 的含义

- $O(g(n))$ 是一个集合函数: 对于 $O(g(n))$ 中的每一个 $h(n)$, 如果存在常数 c 满足, 对于所有的 $n \geq 1$, $h(n) \leq c \cdot g(n)$

例子 $O(g(n))$ 的例子

- $\log n, n \log n, 1000 n^2, 10 n^2 + 1000 n \log n + \sqrt{n}$ 都在 $O(n^2)$ 的复杂度内。
- $O(n) = O(n^2)$, $O(n \log n + \sqrt{n}) = O(n^2)$
- $O(n^2) \neq O(n)$

- 两边大O比较大小
 - n 取比较大的时候判断
- 时间复杂度计算
- 递归树
 - 递归式
 - 有的要展开才能求
 - master定理
 - 课上练习的复杂度分析
- 分治法
 - 元素查找
 - 可能有元素查找算法题, 偏抽象, 解法简单
 - 分治代码和迭代式子
 - 分组比较难, 考试不会考
- 排序
 - 归并排序
 - 考得比较简单
 - 快速排序
 - 考得比较简单
 - 分治思路
 - 考得比较简单
 - 排序算法的时间复杂度表
 - 重点记算法

- 堆排序
简单了解，不重点考察
- 排序的时间复杂度下界表

算法分析与设计 中南大学

排序算法 以非递减顺序排序。

算法	(worst) time	(best) time	(avg) time	方法
MergeSort (归并排序)	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	divide & conquer
QuickSort (快速排序)	n^2	$O(n \log n)$	$O(n \log n)$	divide & conquer
InsertSort (插入排序)	n^2	n	n^2	insertion
SelectSort (选择排序)	n^2	n^2	n^2	selection
BubbleSort (冒泡排序)	n^2	n	n^2	swapping
HeapSort (堆排序)	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	swapping

笔记

记定理，不用会证

- 计数排序

线性时间排序算法

6.1 计数排序 问题和思路

思路：对于每个元素 $A[i]$ ，统计 $A[0..n-1]$ 中不大于 $A[i]$ 的元素数量，表示为 h_i 。在输出结果中， $A[i]$ 应当位于 h_i 的位置上。

计数排序是稳定排序算法。

稳定排序：相同的值在排序过程中一直保持相同的相对顺序。

稳定排序算法：冒泡排序，插入排序，归并排序。

不稳定排序算法：快速排序，堆排序，选择排序。

要求线性时间复杂度的时候用

- 计数排序和基数排序的思路
- 算法稳定性
- 动态规划
不考最长公共子序列
- ~~les~~ 不考
- 算法设计思路
- 动态规划的特点
- 实现动态规划的方法

- 自顶向下
- 自底向上
- ~~0/1背包问题动态规划方法没有~~
- 加权区间调度
 - 动态规划的核心思想
 - 贪心的核心思想
 - 了解思路和核心思想
 - 求有向无环图的最短路径
 - 编辑距离问题
 - ~~今年可能不涉及~~
- **矩阵连乘问题**

计算量大，会考，可以用表存储子问题

 - 思路和解法
 - 给定多个矩阵怎么加括号
- 活动选择问题
 - ~~贪心证明都不需要掌握~~
 - 思路
 - 如何选择
 - 贪心、动态规划的优缺点/区别
 - 动态规划：计算更复杂，但能获得全局最优解
 - 贪心：简单快速，但是局部最优解不一定是全局最优解
 - 贪心的求解思路
 - **贪心算法**
 - 算法设计题考，会判断用贪心还是动态规划
- 哈夫曼编码
- 最小生成树
 - 两种算法作为贪心案例
 - 了解贪心思路
 - 要判断出用贪心思想求解

- 图算法
 - BFS\DFS
 - 简单了解
 - 注意时间复杂度
 - $O(m+n)$ 或 $\max(O(m), O(n))$
 - 有向图的DFS不管
- 最短路径问题
 - 必考
 - 有向图的Bellman-Ford算法
 - 循环 $n-1$ 次
 - 求单源节点最短路径
 - Bellman-Ford
 - SPFA
 - 因为bmf有很多重复计算，所以引入SPFA算法
 - 了解时间复杂度
 - 注意用的数据结构和对应的时间复杂度
 - 迪式算法
- 回溯
 - 分支界限法
 - 子集合
 - 正负 相交集合
 - 连续子序列和问题
 - 动态规划
 - 分治
 - 最长递增子序列
 - 了解
- 分支界限法

分支-限界法

中南大学

16.1 分支-限界法的基本思想

设计思想:

(1) 确定一个合理的界限函数, 并根据界限函数确定目标函数的界[down, up].

(2) 按照广度优先策略搜索问题的解空间树, 在分支节点上依次扩展该节点的所有孩子节点, 分别估算这些孩子节点的目标函数的可能取值, 某孩子节点的目标函数的可能取值超出目标函数的界, 则将其丢弃; 否则, 将其加入待处理节点表(活节点表)中。

(3) 依次从活节点表中选取使目标函数取得极值的终点成为当前扩展节点, 重复上述过程, 直到找到最优解。

会涉及

只需要掌握
广度优先

- 基本思想
- 重点做优化问题
- 确定上下界 () 难点
- 掌握确定思路
 - 上界一般用贪心
 - 下界一般找最优解, 如果在上界之外就不会是最优解
- 0/1背包问题
- 分支界限法
- 贪心找下界
- **所有最短路径问题**

13.1 所有最短路径问题

问题: 给定一个加权图G(可能存在负边), 图信息用邻接矩阵M存储, 求解G中所有节点对(s,t)的最短路径;

(1)

(2)

	a	b	c	d
a	0	∞	3	∞
b	2	0	∞	∞
c	∞	7	0	1
d	6	∞	∞	0

重点掌握算法

- 单源结点
 - 多源结点
 - Floyd-Warshall
 - NP完全性理论
 - 含义
 - 证明思路
 - 证明一个问题是一个NP完全问题
- 了解，证明过程不用掌握**

以上内容整理于 [幕布文档](#)

算法重点

时间复杂度

元素查找

快排

排序时间复杂度比较（比较排序的下界是 $n \log n$ ）

计数排序（线性时间复杂度）

动规：（ppt21-25）

动态规划的特点

矩阵连乘问题

加权区间调度？

贪心：

贪心证明思路

贪心和动态规划优缺点

贪心会有新的题

活动选择问题

图：

最小生成树（prim kruskal）

dfs bfs？ 时间复杂度？

单源最短路径（两种算法 bellman-ford 迪杰斯特拉）

spfa需要掌握？

Johnson算法不考

弗洛伊德算法（用邻接矩阵存储）

连续子序列和（分治、动规）

分支界限：

01背包-分支界限法

np：

证明np完全问题的思路（需要先证明是np问题，再证明可以归约到一个已知的np完全问题）

简答题 有个一个是np。包含关系？ 问题定义？

计算题（上面都包含了）

算法题 动态规划（一定会有）、贪心、查找

很多题先排序再求解