

# 《算法设计与分析》——贪心策略专题特训

桓

2025 年 6 月 27 日

## 第一部分：区间与选择问题

### 问题 1：活动选择问题 (Activity Selection)

给定  $n$  个活动，每个活动都有一个开始时间  $s_i$  和一个结束时间  $f_i$ 。你只有一个会议室，一次只能进行一个活动。请设计一个算法，选择尽可能多的活动，使得这些活动互不冲突。

要求：

1. **贪心选择性质**：清晰描述你的贪心选择是什么。
2. **算法伪代码**：描述完整的算法流程。
3. **正确性证明**：简要说明为什么该贪心策略是正确的（可使用交换论证）。

### 问题 2：区间覆盖问题 (Interval Covering)

给定一个需要被覆盖的闭区间  $[T_{start}, T_{end}]$  和  $n$  个可选的闭区间  $[s_i, f_i]$ 。请设计一个算法，选择最少数量的区间，完全覆盖  $[T_{start}, T_{end}]$ 。如果无法完全覆盖，请说明。

要求：

1. **贪心选择性质**：描述在每一步中，如何选择下一个区间。
2. **算法伪代码**：给出求解该问题的伪代码。
3. **复杂度分析**：分析算法的时间复杂度。

### 问题 3：无重叠区间 (Non-overlapping Intervals)

给定一个区间的集合，找到需要移除区间的最小数量，使剩余区间互不重叠。

要求：

1. **问题转化**：说明此问题如何转化为一个经典的贪心问题。
2. **贪心选择性质**：基于转化后的问题，描述其贪心选择。
3. **算法伪代码**：描述完整的算法流程。

#### 问题 4: 用最少数量的箭引爆气球 (Minimum Arrows to Burst Balloons)

在二维空间中，有一些球形气球，每个气球占据的  $x$  坐标范围为  $[x_{start}, x_{end}]$ 。你可以从  $x$  轴的任意位置垂直向上射箭。一支箭可以射穿所有与它的  $x$  坐标重合的气球。请计算并返回引爆所有气球所需的最小弓箭数。

要求:

1. **贪心选择性质:** 描述如何确定射出第一支箭的位置，以及后续如何决策。
2. **算法伪代码:** 给出求解该问题的伪代码。
3. **正确性证明:** 简要证明你的贪心选择的正确性。

#### 问题 5: 合并区间 (Merge Intervals)

给定一个区间的集合，合并所有重叠的区间。例如，输入  $[[1,3],[2,6],[8,10],[15,18]]$ ，输出  $[[1,6],[8,10],[15,18]]$ 。

要求:

1. **核心思想:** 描述解决此问题的关键预处理步骤和贪心合并的逻辑。
2. **算法伪代码:** 描述完整的算法流程。
3. **复杂度分析:** 分析算法的时间和空间复杂度。

## 第二部分: 基于排序的贪心策略

#### 问题 6: 分数背包问题 (Fractional Knapsack)

有一个背包，最大承重为  $W$ 。有  $n$  个物品，第  $i$  个物品的重量为  $w_i$ ，价值为  $v_i$ 。物品可以被分割成任意小的部分。请设计一个算法，使得装入背包的物品总价值最大。

要求:

1. **贪心选择性质:** 定义用于排序物品的指标，并描述贪心选择。
2. **算法伪代码:** 描述完整的算法流程。
3. **与 0-1 背包对比:** 简要说明为什么贪心法适用于分数背包，而不适用于 0-1 背包问题。

#### 问题 7: 击败怪兽 (Minimize Total Damage)

在某个 VR 游戏中，有  $n$  只怪兽正在袭击村庄。击败第  $i$  只怪兽需要不间断地花费  $T_i$  时间，在此期间，其它未被攻击的怪兽会对村庄造成破坏。第  $j$  只怪兽每个单位时间会对村庄产生  $D_j$  的破坏。勇者希望确定一个击败所有怪兽的顺序，使得村庄受到的总破坏最小。

**要求:**

1. **贪心选择性质:** 描述确定怪兽击败顺序的贪心策略。
2. **算法伪代码:** 给出该贪心算法的伪代码。
3. **正确性证明:** 使用交换论证法证明该策略的最优性。

### 问题 8: 分发饼干 (Assign Cookies)

假设你是一位老师, 手上有一些饼干, 大小记为数组  $s$ 。同时你有一群孩子, 每个孩子  $i$  有一个胃口值  $g_i$ , 表示他至少需要一块大小为  $g_i$  的饼干才能被满足。每块饼干只能分给一个孩子。请设计算法, 尽可能多地满足孩子。

**要求:**

1. **贪心选择性质:** 描述如何匹配孩子和饼干的贪心策略。
2. **算法伪代码:** 给出求解该问题的伪代码。
3. **正确性证明:** 简要证明该策略的正确性。

### 问题 9: 最小初始能量问题 (Minimum Initial Energy)

你有  $n$  个任务需要完成, 每个任务有两个参数  $(a_i, b_i)$ , 分别表示完成该任务需要消耗的能量  $a_i$  和开启该任务所需要的能量阈值  $b_i$  (即你当前持有的能量必须  $\geq b_i$  才能开始任务  $i$ )。完成任务后, 你的能量会减少  $a_i$ 。请计算出要完成全部这  $n$  个任务所需要的最小初始能量。

**要求:**

1. **贪心选择性质:** 描述确定任务执行顺序的贪心策略。
2. **算法伪代码:** 给出该贪心算法的伪代码。
3. **正确性证明:** 使用交换论证法证明该策略的最优性。

### 问题 10: 根据身高重建队列 (Queue Reconstruction by Height)

假设有打乱顺序的一群人站成一个队列。每个人由一个整数对  $(h, k)$  表示, 其中  $h$  是这个人的身高,  $k$  是排在这个人前面且身高大于或等于  $h$  的人数。请你编写一个算法, 根据这个列表重建出原始的队列。

**要求:**

1. **贪心选择性质:** 描述一种排序和插入的策略, 可以逐步构建出正确的队列。
2. **算法伪代码:** 描述完整的算法流程。
3. **正确性分析:** 解释为什么每次贪心插入后, 不会影响之前已插入元素的正确性。

## 第三部分：经典模型与图论应用

### 问题 11: 霍夫曼编码 (Huffman Coding)

给定  $n$  个字符及其出现频率  $\{f_1, f_2, \dots, f_n\}$ 。请为每个字符设计一个二进制前缀码（即任何字符的编码都不是另一个字符编码的前缀），使得编码后的文本总长度最小。

要求：

1. **贪心选择性质**：描述构建霍夫曼树的贪心步骤。
2. **算法伪代码**：描述构建霍夫曼树和生成编码的完整流程。
3. **正确性证明**：简述霍夫曼算法正确性的两个关键引理（贪心选择引理和最优子结构引理）。

### 问题 12: 最小生成树 (Minimum Spanning Tree)

给定一个带权的无向连通图  $G = (V, E)$ ，其中  $V$  是顶点集合， $E$  是边集合，每条边  $(u, v) \in E$  都有一个权重  $w(u, v)$ 。请找到图的一个子图  $T$ ，它是一棵树，连接了所有的顶点，并且其所有边的权重之和最小。

要求：

1. **贪心策略描述**：描述 Prim 算法或 Kruskal 算法中的任意一种的贪心选择性质。
2. **算法伪代码**：写出你所选择算法的伪代码。
3. **正确性分析**：简述该算法为何能保证找到最小生成树（可以引用“切割性质”）。

### 问题 13: 加油站 (Gas Station)

在一条环形路线上有  $n$  个加油站，第  $i$  个加油站有汽油  $\text{gas}[i]$  升。你有一辆油箱无限大的汽车，从第  $i$  个加油站开往第  $i + 1$  个加油站需要消耗  $\text{cost}[i]$  升汽油。你从其中的一个加油站出发，初始时油箱为空。如果你可以绕环路行驶一周，则返回出发时加油站的编号，否则返回 -1。如果解存在，题目保证解是唯一的。

要求：

1. **贪心选择性质**：描述如何判断是否存在解，以及如何寻找起始点的贪心策略。
2. **算法伪代码**：给出求解该问题的高效伪代码。
3. **正确性分析**：证明为什么当总油量大于等于总消耗时一定有解，并解释寻找起点的逻辑。

### 问题 14: 跳跃游戏 (Jump Game)

给定一个非负整数数组 `nums`，你最初位于数组的第一个位置。数组中的每个元素代表你在该位置可以跳跃的最大长度。判断你是否能够到达最后一个位置。

**要求:**

1. **贪心选择性质:** 描述一种贪心策略, 用于判断是否能到达终点。
2. **算法伪代码:** 给出该贪心算法的伪代码。
3. **复杂度分析:** 分析算法的时间和空间复杂度。

### 问题 15: 拼接数字以获得最小值 (Minimum Number by Concatenation)

给定一个正整数数组, 请将数组里所有数字拼接起来排成一个数, 打印能拼接出的所有数字中最小的一个。例如输入数组  $\{3, 32, 321\}$ , 则打印出这三个数字能排成的最小数字为 321323。

**要求:**

1. **贪心选择性质:** 定义一种新的“小于”关系, 用于对数组中的数字 (字符串形式) 进行排序。
2. **算法伪代码:** 描述完整的算法流程。
3. **正确性证明:** 证明你定义的比较规则具有传递性, 从而保证排序的有效性。

## 第四部分: 综合与变种问题

### 问题 16: 移除 K 位数字 (Remove K Digits)

给定一个以字符串表示的非负整数 `num`, 从中移除  $k$  个数字, 使得剩下的数字最小。

**要求:**

1. **贪心选择性质:** 描述在遍历数字时, 如何决定是否移除当前数字或之前的数字。
2. **核心数据结构:** 说明哪种数据结构适合实现这个贪心策略。
3. **算法伪代码:** 给出完整的算法伪代码, 并处理好前导零和移除数量不足等边界情况。

### 问题 17: 分糖果 (Candy)

$n$  个孩子站成一排, 给定一个整数数组 `ratings` 代表每个孩子的评分。你需要按照以下要求, 给这些孩子分发糖果:

- 每个孩子至少分配到 1 个糖果。
- 相邻两个孩子评分更高的孩子会获得更多的糖果。

请计算最少需要准备的糖果数目。

**要求:**

1. **贪心策略:** 描述一个可以解决此问题的贪心策略, 可能需要多次遍历。
2. **算法伪代码:** 给出该策略的伪代码。
3. **正确性分析:** 解释为什么该策略 (例如, 左右两次遍历) 能够满足所有约束条件并得到最优解。

## 问题 18: 任务调度器 (Task Scheduler)

给你一个用字符数组 `tasks` 表示的 CPU 需要执行的任务列表，其中每个字母代表一种不同种类的任务。任务可以以任意顺序执行，并且每个任务都可以在一个单位时间内执行完。在任何一个单位时间，CPU 可以执行一个任务，或者处于待命状态。但是，两个相同种类的任务之间必须有长度为  $n$  的冷却时间。返回完成所有任务所需要的 **最短时间**。

**要求:**

1. **贪心选择性质:** 描述在每个时间点，应该优先安排哪种任务。
2. **核心思想:** 分析最短时间的构成，并基于此推导出一个数学公式或构造方法。
3. **算法流程:** 描述计算最短时间的具体步骤。

## 问题 19: 划分字母区间 (Partition Labels)

一个字符串  $S$  由小写字母构成。我们希望把这个字符串划分为尽可能多的片段，同一个字母只会出现在其中的一个片段。返回一个表示每个字符串片段的长度的列表。

**要求:**

1. **贪心选择性质:** 描述如何确定第一个划分片段的结束位置，以及如何继续划分剩余部分。
2. **算法伪代码:** 给出求解该问题的伪代码。
3. **复杂度分析:** 分析算法的时间和空间复杂度。

## 问题 20: 买卖股票的最佳时机 II (Best Time to Buy and Sell Stock II)

给定一个数组 `prices`，其中 `prices[i]` 是一支给定股票第  $i$  天的价格。设计一个算法来计算你能获取的最大利润。你可以尽可能地完成更多的交易（多次买卖一支股票）。注意：你不能同时参与多笔交易（你必须在再次购买前出售掉之前的股票）。

**要求:**

1. **贪心选择性质:** 描述一个极其简单的贪心策略，用于累积利润。
2. **算法伪代码:** 给出该贪心算法的伪代码。
3. **正确性分析:** 解释为什么这个看似简单的策略等价于找到所有上升子区间的利润之和。