数据结构与算法

Data Structure and Algorithm

XXVII. 贪心算法

授课人：Kevin Feng
翻　译：王　落　桐

课前回顾

- 数据结构及算法
- 数学回顾
- 数组（Array）和数组列表（Array List）
- 递归 vs. 迭代
- 二分法搜索
- 分治法
- 链表
- 栈和队列

课前回顾

- 哈希（Hash）
- 树（Tree）
- 堆（Heap）
- 图论（Graph）
- 双向指针（Two Pointers）
- 滑动窗口（Sliding Window）
- 动态规划（Dynamic Programming）
- 比特操作（Bit Manipolation）

贪心算法 | Greedy

- 在对问题求解时，总是做出在当前看来是最好的选择
- 基本思路：
  - 建立数学模型来描述问题。
  - 把求解的问题分成若干个子问题。
  - 对每一子问题求解，得到子问题的局部最优解。
  - 把子问题的解局部最优解合成原来解问题的一个解。
- 贪心策略适用的前提是：局部最优策略能导致产生全局最优解。
- 基本要素：
  - 贪心选择性质：所求问题的整体最优解可以通过一系列局部最优的选择 （贪心算法与动态规划算法的主要区别，动态规划算法通常以自底向上的方式解各子问题，而贪心算法则通常以自顶向下的方式进行，以迭代的方式作出相继的贪心选择，每作一次贪心选择就将所求问题简化为规模更小的子问题。）
  - 当一个问题的最优解包含其子问题的最优解时，称此问题具有最优子结构性质。问题的最优子结构性质是该问题可用动态规划算法或贪心算法求解的关键特征。

我们遇到过的贪心问题 Greedy Problems We Have Seen

- Dijkstra Algorithm
- Bellman - Ford Algorithm

Greedy vs. Dynamic Programming

Greedy	***Dynamic Programming***
A greedy algorithm is one that at a given point in time, makes a local optimization.	Dynamic programming can be thought of as 'smart' recursion.,It often requires one to break down a problem into smaller components that can be cached.
Greedy algorithms have a local choice of the subproblem that will lead to an optimal answer	Dynamic programming would solve all dependent subproblems and then select one that would lead to an optimal solution.
A greedy algorithm is one which finds optimal solution at each and every stage with the hope of finding global optimum at the end.	A Dynamic algorithm is applicable to problems that exhibit Overlapping subproblems and Optimal substructure properties.
More efficient as compared,to dynamic programming	Less efficient as compared to greedy approach

找硬币：

- 设有n 种不同面值的硬币，各硬币的面值存于数组T〔1:n 〕中。现要用这些面值的硬币来找钱。可以使用的各种面值的硬币面值{1，2，5，10，20，50，100，500，1000} 对任意钱数0≤m≤20001，设计一个用最少硬币找钱 的方法。

活动问题

- N个活动，每个活动的开始时间为si，结束时间是fi。如果 si ≥ fj or sj ≥ fi 则可以定义这两场活动不冲突。试着找到一种能够找到最多的非冲突活动的集合（S）。也就是说无法找到集合S’ 使得|S’| > |S|。

最小的数字问题

- 如何找到给定数字总和s和位数m的最小数字？
- 输入：s = 9，m = 2
- 输出：18
- 还有许多其他可能的数字，如45，54，90等，数字总和为9，数字位数为2.
  其中最小的为18。
- 输入：s = 20，m = 3
- 输出：299

两个数字的最小和

- 给定一个数字数组(数值从0到9)，找出由数组数字形成的两个数字的最小可能和。给定数组的所有数字必须用于形成两个数字。
- 输入：[6，8，4，5，2，3]
- 输出：604
- 最小总和由数字组成
- 358和246
- 输入：[5，3，0，7，4]
- 输出：82
- 最小总和由数字组成
- 35和047

以最低的成本连接绳索

- 有n条不同长度的绳索，我们需要将这些绳索连接成一根绳子。连接两条绳索的成本等于它们长度的总和。我们需要以最低的成本连接绳索。
- 例如，如果我们获得4条长度为4，3，2和6的绳索，我们可以通过以下方式连接绳索。
  1）首先连接长度为2和3的绳索。现在我们有三根长度为4，6和5的绳索。
  2）现在连接长度为4和5的绳索。现在我们有两根长度为6和9的绳索。
  3）最后连接两条绳索，所有绳索已连接。
- 连接所有绳索的总成本为5+9+15=29。

最小平台数

- 根据所有到达火车站的列车的到达和离开时间，找到火车站所需的最少数量的平台，以免列车等待。
- 我们给出了代表停止列车到达和离开时间的两个数组
- 例子：
- 输入：arr [] = {9:00, 9:40, 9:50, 11:00, 15:00, 18:00}
- dep [] = {9:10, 12:00, 11:20, 11:30, 19:00, 20:00}
- 输出：3
- 一次最多有三班列车（时间为11:00至11:20）

部分背包问题

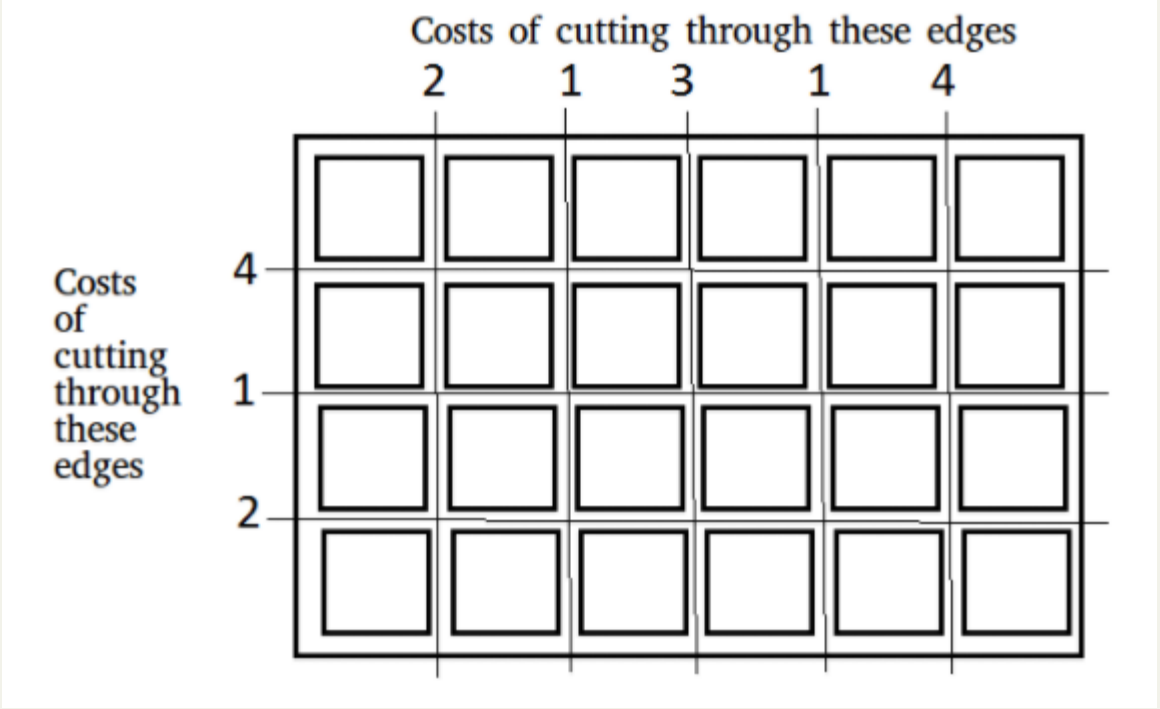
- 给定n个项目的权重和值，我们需要把这些项目放入W的背包中，以获得背包中最大的总价值。
- 在0-1背包问题中，我们不允许分解物品。我们要么拿整个项目，要么不拿。
- 在分数背包中，我们可以打破物品以最大化背包的总价值。这个问题，我们可以打破项目也被称为分数背包问题。

分蛋糕

- 有N个孩子，M个蛋糕。
- 每个蛋糕有自己的尺寸
- 每个孩子对蛋糕的尺寸有个最小要求，这个孩子接受这个蛋糕，那么尺寸必须达标
- 那么现在，你的任务就是如何分配这些蛋糕，使得尽量多的孩子接受你的蛋糕。。

将板子切割成正方形的最小成本

- 给定一个长度为m和宽度为n的电路板，我们需要将这个电路板分成m*n个正方形，使得断开成本最小。每个板的切割成本将提供给电路板。总之，我们需要选择这样的一系列切割，以使成本最小化。



字典中最小的数组

- 给定一个数组arr[ ]，找到在最多K次连续交换之后可以获得的字典顺序最小的数组。
- 输入：
- arr = [7,6,9,2,1], k = 3
- 输出：arr = [2,7,6,9,1]
- 说明：

  数组是：7，6，9，2，1

  交换1：7，6，2，9，1

  交换2：7，2，6，9，1

  交换3：2，7，6，9，1

  所以我们在k = 3交换后的最后一个数组：

  2，7，6，9，1

最小最大高度差

- 给定n个塔的高度和一个k值，我们需要在k>0的情况下用k(仅一次)来增加或降低每个塔的高度。任务是最小化修改后最长和最短塔高度之间的差值，并输出这个差值。
- 输入：arr[ ] = {1，15，10}，k =6
- 输出：最大差值为5。
- 说明：我们将1更改为6，15更改为9和10更改为4。最大差值为5(介于4和9之间)。我们不能得到更低的差额。

数据结构与算法

Data Structure and Algorithm

XXVII. 贪心算法结束

授课人：Kevin Feng
翻　　译：王落桐