GREEDY ALGORITHM

20数媒网络 潘东逸杰

贪心算法（英语：greedy algorithm），又称贪婪算法，是一种在每一步选择中都采取在当前状态下最好或最优（即最有利）的选择，从而希望导致结果是最好或最优的算法。

-Wiki

贪心的局限性

- 可想而知，并不是所有的时候贪心法都能获得最优解，所以一般用贪心法的时候，都要确保自己能证明其正确性。
- 贪心算法在有最优子结构的问题中尤为有效。最优子结构的意思是问题能够分解成子问题来解决。子问题的最优解能递推到最终问题的最优解。

贪心法的证明方法

- 贪心算法有两种证明方法，反证法和归纳法。一般情况下，一道题只会用到其中一种方法来证明。

反证法（CONTRADICTION)

- 先假设命题的结论不成立，由此经过推理得出矛盾，由矛盾判定所作假设不正确，从而得到原命题成立，这种方法叫反证法。

举个例子：欧几里得素数定理

- 如何证明素数有无限多个？
- 《几何原本》中的经典问题
- 抓一个同学现场证明

反证法的一些具体例子

- 相邻两项交换？范围缩放？具体看例题

数学归纳法（MATHEMATICAL INDUCTION）

- 数学归纳法（Mathematical Induction, MI）是一种数学证明方法，通常被用于证明某个给定命题在整个（或者局部）自然数范围内成立。除了自然数以外，广义上的数学归纳法也可以用于证明一般良基结构，例如：集合论中的树。这种广义的数学归纳法应用于数学逻辑和计算机科学领域，称作结构归纳法。

数学归纳法（MATHEMATICAL INDUCTION）

- 最简单和常见的数学归纳法是证明当n等于任意一个自然数时某命题成立。证明分下面两步：

- 证明当n= 1时命题成立。
- 假设n=m时命题成立，那么可以推导出在n=m+1时命题也成立。（m代表任意自然数）

数学归纳法（MATHEMATICAL INDUCTION）

➤ 这种方法的原理在于：首先证明在某个起点值时命题成立，然后证明从一个值到下一个值的过程有效。当这两点都已经证明，那么任意值都可以通过反复使用这个方法推导出来。把这个方法想成多米诺效应也许更容易理解一些。例如：你有一列很长的直立着的多米诺骨牌，如果你可以：

- 证明第一张骨牌会倒。
- 证明只要任意一张骨牌倒了，那么与其相邻的下一张骨牌也会倒。

数学归纳法（MATHEMATICAL INDUCTION）

- **23.**（本小题满分 10 分）

  设 $n\in\mathbf{N}^*$，对 1，2，…，n 的一个排列 $i_1i_2\cdots i_n$，如果当 $s<t$ 时，有 $i_s>i_t$，则称 (i_s,i_t) 是排列 $i_1i_2\cdots i_n$ 的一个逆序，排列 $i_1i_2\cdots i_n$ 的所有逆序的总个数称为其逆序数. 例如：对 1，2，3 的一个排列 231，只有两个逆序 (2，1)，(3，1)，则排列 231 的逆序数为 2. 记 $f_n(k)$ 为 1，2，…，n 的所有排列中逆序数为 k 的全部排列的个数.

  (1) 求 $f_3(2)$，$f_4(2)$ 的值；

  (2) 求 $f_n(2)(n\geqslant 5)$ 的表达式（用 n 表示）.

- 江苏高考2018.数学.T23压轴题

数学归纳法（MATHEMATICAL INDUCTION）

➤ 23.【必做题】本小题主要考查计数原理、排列等基础知识，考查运算求解能力和推理论证能力. 满分 10 分.

解：(1) 记 $\tau(abc)$ 为排列 abc 的逆序数，对 1，2，3 的所有排列，有

$\tau(123)=0$，$\tau(132)=1$，$\tau(213)=1$，$\tau(231)=2$，$\tau(312)=2$，$\tau(321)=3$，

所以 $f_3(0)=1$，$f_3(1)=f_3(2)=2$.

对 1，2，3，4 的排列，利用已有的 1，2，3 的排列，将数字 4 添加进去，4 在新排列中的位置只能是最后三个位置.

因此，$f_4(2)=f_3(2)+f_3(1)+f_3(0)=5$.

(2) 对一般的 $n(n\geqslant 4)$ 的情形，逆序数为 0 的排列只有一个：$12\cdots n$，所以 $f_n(0)=1$.

逆序数为 1 的排列只能是将排列 $12\cdots n$ 中的任意相邻两个数字调换位置得到的排列，所以 $f_n(1)=n-1$.

为计算 $f_{n+1}(2)$，当 $1，2，\cdots，n$ 的排列及其逆序数确定后，将 $n+1$ 添加进原排列，$n+1$ 在新排列中的位置只能是最后三个位置.

因此，$f_{n+1}(2)=f_n(2)+f_n(1)+f_n(0)=f_n(2)+n$.

当 $n\geqslant 5$ 时，

$$f_n(2)=[f_n(2)-f_{n-1}(2)]+[f_{n-1}(2)-f_{n-2}(2)]+\cdots+[f_5(2)-f_4(2)]+f_4(2)$$

$$=(n-1)+(n-2)+\cdots+4+f_4(2)=\frac{n^2-n-2}{2},$$

因此，当 $n\geqslant 5$ 时，$f_n(2)=\dfrac{n^2-n-2}{2}$.

➤ 江苏高考2018.数学.T23压轴题

贪心的一些性质

- 每次决策的时候采用当前的最优策略
- 问题的整体最优性能从局部最优性导出
- 无后效性，即每个状态，以后的过程不会影响以前的状态，只和当前状态有关。

贪心的范式

- 算法设计的关键是贪心决策的选择
- 1.建立数学模型来描述问题
- 2.把需要求解的问题划分为若干个子问题
- 3.对每一个子问题，求出局部最优解
- 4.将子问题的解的组合，得到全局最优解

贪心的范式

- 话说得很玄乎，一套一套的。
- 但是真正做题时，很多时候，贪心就是一种“猜”。
- 我“猜”这么贪心是正确的，并能大概想明白为什么是正确的，我就可以写了。
- 也许正确，也许错误。
- 但错误积累多了，经验丰富了，错的也就自然而然的变成对的了。
- 一言以蔽之，多做多练多尝试。

钱币找零问题

【问题】 有面值1元、5元、10元、20元、50元和100元的纸币各若干张，支付固定金额k元。 求最少需要多少张纸币？

怎么设计算法？想一想。

部分背包問題

【问题】 给定一个最大负重为m的背包和n种物品。第i种物品重量wi，价格vi。 求能使背包中放入物品最大的价格之和。

怎么设计算法？想一想。

活动时间安排问题

【问题】 设有N个活动时间集合，每个活动都要使用同一个资源，比如说会议场，而且同一时间内只能有一个活动使用，每个活动都有一个使用活动的开始si和结束时间fi，即他的使用区间为（si,fi）,现在要求你分配活动占用时间表，即哪些活动占用该会议室，哪些不占用，使得他们不冲突，要求是尽可能多的使参加的活动最大化，即所占时间区间最大化！

怎么设计算法？想一想。

活动时间安排问题

i	1	2	3	4	5	6	7	8	9	10	11
S[i]	1	3	0	5	3	5	6	8	8	2	12
f[i]	4	5	6	7	8	9	10	11	12	13	14

怎么设计算法？想一想。

数字组合问题

设有N个正整数，现在需要你设计一个程序，使他们连接在一起成为最大的数字，例3个整数 12,456,342 很明显是45634212为最大，4个整数 342，45,7,98 显然为98745342最大

怎么设计算法？想一想。

国王游戏

> 恰逢 H 国国庆，国王邀请 n 位大臣来玩一个有奖游戏。首先，他让每个大臣在左、右手上面分别写下一个整数，国王自己也在左、右手上各写一个整数。然后，让这 n 位大臣排成一排，国王站在队伍的最前面。排好队后，所有的大臣都会获得国王奖赏的若干金币，每位大臣获得的金币数分别是：**排在该大臣前面的所有人的左手上的数的乘积除以他自己右手上的数，然后向下取整得到的结果。**
>
> 国王不希望某一个大臣获得特别多的奖赏，所以他想请你帮他重新安排一下队伍的顺序，使得获得奖赏最多的大臣，所获奖赏尽可能的少。注意，国王的位置始终在队伍的最前面。

国王游戏

输入格式

第一行包含一个整数 n，表示大臣的人数。

第二行包含两个整数 a 和 b，之间用一个空格隔开，分别表示国王左手和右手上的整数。

接下来 n 行，每行包含两个整数 a 和 b，之间用一个空格隔开，分别表示每个大臣左手和右手上的整数。

输出格式

输出只有一行，包含一个整数，表示重新排列后的队伍中获奖赏最多的大臣所获得的金币数。

数据范围

对于 20% 的数据，有 $1 \leq n \leq 10$，$0 < a$、$b < 8$；

对于 40% 的数据，有 $1 \leq n \leq 20$，$0 < a$、$b < 8$；

对于 60% 的数据，有 $1 \leq n \leq 100$；

对于 60% 的数据，保证答案不超过 10^9；

对于 100% 的数据，有 $1 \leq n \leq 1,000$，$0 < a$、$b < 10000$。

例题

数据范围

对于 20% 的数据，有 $1 \le n \le 10$，$0 < a$、$b < 8$；

对于 40% 的数据，有 $1 \le n \le 20$，$0 < a$、$b < 8$；

对于 60% 的数据，有 $1 \le n \le 100$；

对于 60% 的数据，保证答案不超过 10^9；

对于 100% 的数据，有 $1 \le n \le 1,000$，$0 < a$、$b < 10000$。

样例说明

按 1、2、3 号大臣这样排列队伍，获得奖赏最多的大臣所获得金币数为 2；

按 1、3、2 这样排列队伍，获得奖赏最多的大臣所获得金币数为 2；

按 2、1、3 这样排列队伍，获得奖赏最多的大臣所获得金币数为 2；

按 2、3、1 这样排列队伍，获得奖赏最多的大臣所获得金币数为 9；

按 3、1、2 这样排列队伍，获得奖赏最多的大臣所获得金币数为 2；

按 3、2、1 这样排列队伍，获得奖赏最多的大臣所获得金币数为 9。

因此，奖赏最多的大臣最少获得 2 个金币，答案输出 2。

例题

Sample Input

```
3
1 1
2 3
7 4
4 6
```

Sample Output

```
2
```

一些个讲解

怎么贪心啊这题?
如何使用高精度?
面向对象的基本概念?

THANKS