贪心算法

贪心算法（又称贪婪算法）是指，在对[问题求解](https://baike.baidu.com/item/%E9%97%AE%E9%A2%98%E6%B1%82%E8%A7%A3/6693186" \t "https://baike.baidu.com/item/%E8%B4%AA%E5%BF%83%E7%AE%97%E6%B3%95/_blank)时，总是做出在当前看来是最好的选择。也就是说，不从整体最优上加以考虑，算法得到的是在某种意义上的局部[最优解](https://baike.baidu.com/item/%E6%9C%80%E4%BC%98%E8%A7%A3/5208902" \t "https://baike.baidu.com/item/%E8%B4%AA%E5%BF%83%E7%AE%97%E6%B3%95/_blank)。

**背包问题**

有一个背包，背包容量是M=150kg。有7个物品，物品不可以分割成任意大小。要求尽可能让装入背包中的物品总价值最大，但不能超过总容量。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 物品 | A | B | C | D | E | F | G |
| 重量 | 35 | 30 | 6 | 50 | 40 | 10 | 25 |
| 价值 | 10 | 40 | 30 | 50 | 35 | 40 | 30 |

分析：

[目标函数](https://baike.baidu.com/item/%E7%9B%AE%E6%A0%87%E5%87%BD%E6%95%B0" \t "https://baike.baidu.com/item/%E8%B4%AA%E5%BF%83%E7%AE%97%E6%B3%95/_blank)：∑pi最大 [1]

约束条件是装入的物品总重量不超过背包容量：∑wi<=M(M=150）

⑴根据贪心的策略，每次挑选价值最大的物品装入背包，得到的结果是否最优？

⑵每次挑选所占重量最小的物品装入是否能得到最优解？

⑶每次选取单位重量价值最大的物品，成为解本题的策略。

**马踏棋盘**

在8×8方格的棋盘上，从任意指定方格出发，为马寻找一条走遍棋盘每一格并且只经过一次的一条路径。

分析：

优先选择‘出口’最小的进行搜索，‘出口’的意思是在这些子结点中它们的可行子结点的个数，也就是‘孙子’结点越少的越优先跳，为什么要这样选取，这是一种局部调整最优的做法，如果优先选择出口多的子结点，那出口少的子结点就会越来越多，很可能出现‘死’结点（顾名思义就是没有出口又没有跳过的结点）

**如把3/7和13/23分别化为三个[单位分数](https://baike.baidu.com/item/%E5%8D%95%E4%BD%8D%E5%88%86%E6%95%B0" \t "https://baike.baidu.com/item/%E8%B4%AA%E5%BF%83%E7%AE%97%E6%B3%95/_blank)的和。**

步骤一： 用b 除以a，得商数q1 及余数r1。（r1=b - a\*q1）

步骤二：把a/b 记作：a/b=1/(q1+1）+(a-r1)/b(q1+1）

步骤三：重复步骤2，直到分解完毕

3/7=1/3+2/21=1/3+1/11+1/231

13/23=1/2+3/46=1/2+1/16+1/368

**均分纸牌**

有 N 堆纸牌，编号分别为 1，2，…, N。每堆上有若干张，但纸牌总数必为 N 的倍数。可以在任一堆上取若于张纸牌，然后移动。

　　移牌规则为：在编号为 1 堆上取的纸牌，只能移到编号为 2 的堆上；在编号为 N 的堆上取的纸牌，只能移到编号为 N-1 的堆上；其他堆上取的纸牌，可以移到相邻左边或右边的堆上。

　　现在要求找出一种移动方法，用最少的移动次数使每堆上纸牌数都一样多。

　　例如 N=4x4 堆纸牌数分别为：

　　①　9　②　8　③　17　④　6

　　移动3次可达到目的：

从 ③ 取 4 张牌放到 ④ （9 8 13 10）

从 ③ 取 3 张牌放到 ②（9 11 10 10）

从 ② 取 1 张牌放到①（10 10 10 10）。

分析：

从第一堆牌开始处理，如果第一堆牌整好是avg那么就放在一边不管了。

如果第一堆牌不是avg，那么就要把第二堆牌（合法的移动只有从2移到1，这也是这个算法的精髓之处）移动几张到第一堆，恰好使第一堆等于avg，从而只考虑第二堆开始到第N堆为止这些堆如何搞的子问题。然后依次递归下去。

这里的一个小技巧是认为牌数可以为负数，这样才能继续下去。综上，这个步骤是合理的。但是看不出来是最优的。可见，贪心法确实是比较容易实现，因为比较符合人类直觉，但是不好证明。

 