

贪心算法

题目：Assign Cookies

语言：python3

英文版链接：<https://leetcode.com/problems/assign-cookies/description/>

中文版链接：<https://leetcode-cn.com/problems/assign-cookies/>

题目分析

贪心算法是指，在对问题求解时，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，他所做出的是在某种意义上的局部最优解。

那么对于本题来说，饼干应当在能满足的基础上尽量小，这样大饼干就能满足胃口更大的孩子。

证明：假设在某次选择中，贪心策略选择给当前满足度最小的孩子分配第 m 个饼干，第 m 个饼干为可以满足该孩子的最小饼干。假设存在一种最优策略，给该孩子分配第 n 个饼干，并且 $m < n$ 。我们可以发现，经过这一轮分配，贪心策略分配后剩下的饼干一定有一个比最优策略来得大。因此在后续的分配中，贪心策略一定能满足更多的孩子。也就是说不存在比贪心策略更优的策略，即贪心策略就是最优策略。

贪心算法适用的条件

问题的求解可以由一系列的决策步骤构成，每步决策依赖于某种局部最优的贪心策略。正确的贪心策略要保证每一步基于局部优化性质的选择最终导致全局的最优解。

如果不具有上述性质，贪心法对某些实例只能得到近似解。

主要设计步骤

将问题的求解看作是一系列的决策；
确定每一步决策所依据的局部优化性质；
证明每一步基于局部优化性质的选择最终导致全局最优解。

贪心法正确性证明常用方法

数学归纳法
交换论证法

注意

贪心法虽然设计简单，但一定要证明其正确性！

答案

```
class Solution:
    def findContentChildren(self, g, s):
        g = sorted(g)
        s = sorted(s)
        cnt_g = 0
        cnt_s = 0
        while cnt_g < len(g) and cnt_s < len(s):
            if g[cnt_g] <= s[cnt_s]:
                # 饼干大于等于人的胃口，满足人的胃口，人数加一
                cnt_g += 1
            # 不管满不满足，饼干都耗费了一个
            cnt_s += 1
        return cnt_g
```