

漫画：有趣的扔鸡蛋问题

2018-06-25 19:31:13 分类：算法与数据结构 (/category/TheAlgorithm/)

来自：程序员小灰 (<https://mp.weixin.qq.com/s/nMC55qvgsQNQfncAEOM20Q>)
(微信号：chengxuyuanxiaohui)，作者：小灰

老板，我明天请假去看世界杯，
我最喜欢施瓦辛格了！



最新文章

- 1 2017年6月TIOBE...
- 2 哪行工资最高？2...
- 3 美团点评宣布启...
- 4 被刘强东刷屏了...
- 5 致准备报考计算...



(/)

好的，去吧去吧。



———— 第二天 ————

小灰是吧？请简单介绍一下你自己。





(/)

好的!

blah blah blah



下面考你一道智力题：有两个鸡蛋，
从一百层楼上往下扔，以此来测试
鸡蛋的硬度



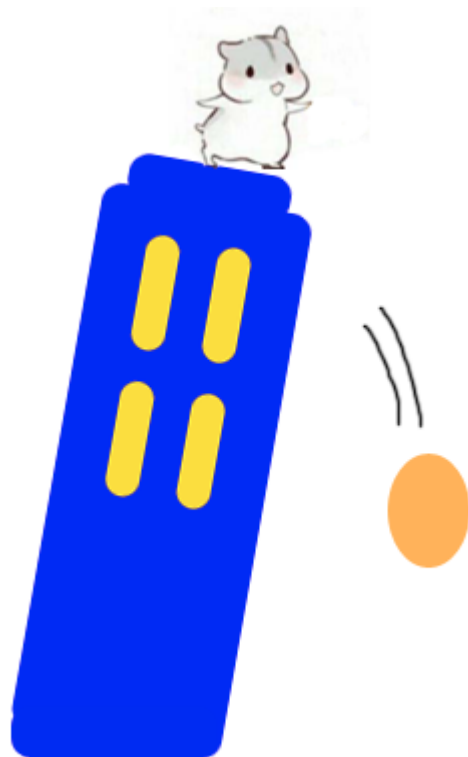
题目：扔鸡蛋问题

有**2**个鸡蛋，从**100**层楼上往下扔，以此来测试鸡蛋的硬度。
比如鸡蛋在第**9**层没有摔碎，在第**10**层摔碎了，那么鸡蛋不会摔碎的临界点就是**9**层。

问：如何用最少的尝试次数，测试出鸡蛋不会摔碎的临界点？



(/)



举个例子，最笨的测试方法是什么样呢？

把其中一个鸡蛋从第1层开始往下扔。

如果在第1层没碎，换到第2层扔

如果在第2层没碎，换到第3层扔

.....

如果第59层没碎，换到第60层扔

如果第60层碎了，说明不会摔碎的临界点是第59层

在最坏情况下，这个方法需要扔100次。



(/)

还真是浪费粮食呀，
让我想想……



有了！我可以用二分法，把
第一枚鸡蛋从 50 层楼往下
扔，这样可以减少一半次数！



方法一：二分法

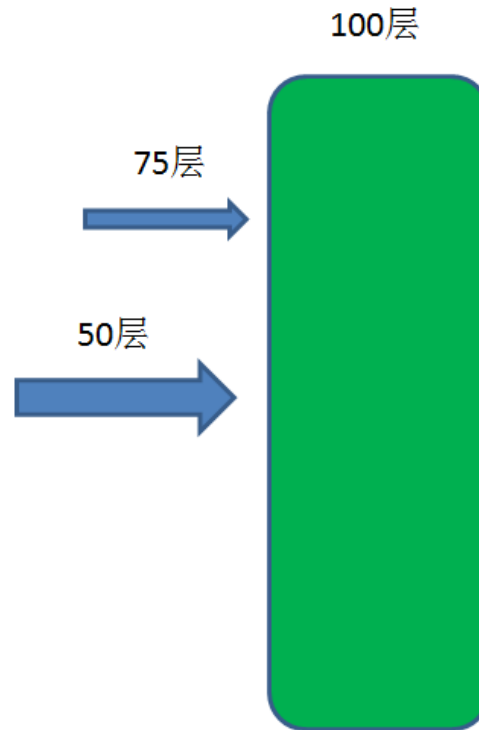
采用类似于二分查找的方法，把鸡蛋从一半楼层（50层）往下扔。

如果第一枚鸡蛋在50层碎了，第二枚鸡蛋就从第1层开始扔，一层一层增长，一直扔到第49层。



如果第一枚鸡蛋在50层没碎了，则继续使用二分法，在剩余楼层的一半（75层）往下扔.....

这个方法在最坏情况下，需要尝试50次。





(/)

OK，这样确实比最笨的方法
优化了一些，还有没有更好
的方法？



呃，那我再想想





有了!100 的开方是 10, 我
(/) 可以每 10 层扔一次



方法二：平方根法

如何让第一枚鸡蛋和第二枚鸡蛋的尝试次数尽可能均衡呢？

很简单，做一个平方根运算，100的平方根是10。

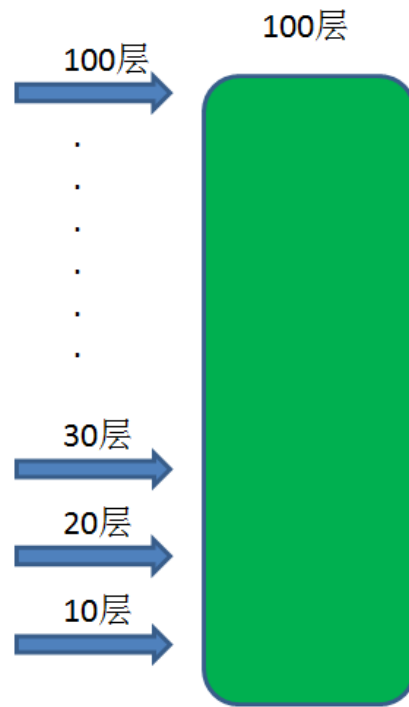
因此，我们尝试每10层扔一次，第一次从10层扔，第二次从20层扔，第三次从30层.....一直扔到100层。

这样的最好情况是在第10层碎掉，尝试次数为 $1 + 9 = 10$ 次。

最坏的情况是在第100层碎掉，尝试次数为 $10 + 9 = 19$ 次。



(/)



不过，这里有一个小小的优化点，我们可以从15层开始扔，接下来从25层、35层扔……一直到95层。

这样最坏情况是在第95层碎掉，尝试次数为 $9 + 9 = 18$ 次。



(/)

OK，这样又优化了很多。
还有没有更优化的方法？



啊，还能再优化吗？
实在想不出更好的方法了……





(/)

呵呵，没关系，回家等通知去吧。



—————

小灰，听说你去面试了？
结果怎么样？





(/)

哎.....



大黄，教教我怎么从楼上
扔鸡蛋呗？





(/)

哈哈，这是谷歌公司出的一道经典面试题。想要直接思考出扔鸡蛋的方式，其实是比较困难的。



我们不妨参考一位著名律师所说的话：把思路逆转过来！





可是，如何逆转我们的思路呢？





(/)

我们不妨这样思考：假设问题存在最优解，这个最优解的最坏情况尝试次数是 x 次。那么，我们第一次扔鸡蛋该选择哪一层？



选择哪一层呢？不知道呀.....



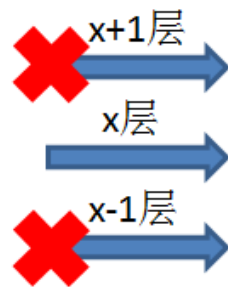


(/)

恰恰是从第 x 层开始扔，选择更高一层或是更低一层都不合适。



100层





假设最优的尝试次数的 x 次，为什么第一次扔就要选择第 x 层呢？

这里的解释会有些烧脑，请小伙伴们坐稳扶好：

假设第一次扔在第 $x+1$ 层：

如果第一个鸡蛋碎了，那么第二个鸡蛋只能从第1层开始一层一层扔，一直扔到第 x 层。

这样一来，我们总共尝试了 $x+1$ 次，和假设尝试 x 次相悖。由此可见，第一次扔的楼层必须小于 $x+1$ 层。

假设第一次扔在第 $x-1$ 层：

如果第一个鸡蛋碎了，那么第二个鸡蛋只能从第1层开始一层一层扔，一直扔到第 $x-2$ 层。

这样一来，我们总共尝试了 $x-2+1 = x-1$ 次，虽然没有超出假设次数，但似乎有些过于保守。

假设第一次扔在第 x 层：

如果第一个鸡蛋碎了，那么第二个鸡蛋只能从第1层开始一层一层扔，一直扔到第 $x-1$ 层。

这样一来，我们总共尝试了 $x-1+1 = x$ 次，刚刚好没有超出假设次数。

因此，要想尽量楼层跨度大一些，又要保证不超过假设的尝试次数 x ，那么第一次扔鸡蛋的最优选择就是第 x 层。



好像明白一些了... 如果第一次扔
(/)鸡蛋是在第 x 层, 并且没有打碎,
那么第二次扔鸡蛋是在哪一层呢?



如果第一次扔鸡蛋没有碎, 我们的尝试
次数消耗了一次, 问题就转化成了两个
鸡蛋在 $100-x$ 层楼往下扔, 要求尝试次
数不得超过 $x-1$ 次。





(/)

所以，在这种情况下，你觉得第二个鸡蛋应该选择哪一层呢？



我知道了，由于尝试次数上限变成了 $x-1$ 次，所以第二次尝试的楼层跨度也是 $x-1$ 层，绝对楼层是 $x+x-1$ 层！





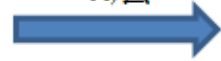
同理，如果鸡蛋还是没有碎，
(/)第三次楼层跨度是 $x-2$ ，第四次
楼层跨度是 $x-3$!





(/)

100层

 $x+x-1+x-2$ 层 $x+x-1$ 层 x 层



(/)

完全正确！根据咱们的总结，可以列出一个楼层数的方程式：



方法三：解方程法

$$x + (x-1) + (x-2) + \dots + 1 = 100$$

这个方程式不难理解：

左边的多项式是各次扔鸡蛋的楼层跨度之和。由于假设尝试x次，所以这个多项式共有x项。

右边是总的楼层数100。

下面我们来解这个方程：

$$x + (x-1) + (x-2) + \dots + 1 = 100 \quad \text{转化为}$$

$$(x+1)*x/2 = 100$$

最终x向上取整，得到 **x = 14**

因此，最优解在最坏情况的尝试次数是**14次**，第一次扔鸡蛋的楼层也是**14层**。



最后，让我们把第一个鸡蛋没碎的情况下，所尝试的楼层数完整列举出来：

14, 27, 39, 50, 60, 69, 77, 84, 90, 95, 99, 100

举个栗子验证下：

假如鸡蛋不会碎的临界点是65层，那么第一个鸡蛋扔出的楼层是14, 27, 50, 60, 69。这时候啪的一声碎了。

第二个鸡蛋继续，从61层开始，61, 62, 63, 64, 65, 66，啪的一声碎了。

因此得到不会碎的临界点65层，总尝试次数是 $6 + 6 = 12 < 14$ 。

明白了，这还真是一个有意思的题目！



嘿嘿，谁要是能在第一次遇到
此题的时候独立想出最佳答
案，智商起码在 140 以上。



其实这道题目还有一个问题：
总共有 M 层楼， N 个鸡蛋，要
找到临界点，需要尝试几次？





(/) 啊，这个扩展问题也太变态了吧.....



要想求出问题的通解，我们需要利用「动态规划」的方法。具体实现我们将会在下期详细介绍。





(/)

好了，关于扔鸡蛋问题我们今天
就介绍到这里，感谢大家的支持！



几点补充：

1. 下一期小灰将会讲解如何利用**动态规划**求出扔鸡蛋问题的通解，不太了解动态规划的小伙伴可以看看小灰之前的漫画预习下：

漫画：什么是动态规划？（整合版）

([http://mp.weixin.qq.com/s?](http://mp.weixin.qq.com/s?__biz=MzIxMjE5MTE1Nw==&mid=2653190796&idx=1&sn=2bf42e5783f3efd03bfb0ecd3cbbc380&chksm=8c990856bbee8140055c3429f59c8f46dc05be20b859f00fe8168efe1e6a954fdc5cfc7246b0&scene=21#wechat_redirect)

[__biz=MzIxMjE5MTE1Nw==&mid=2653190796&idx=1&sn=2bf42e5783f3efd03bfb0ecd3cbbc380&chksm=8c990856bbee8140055c3429f59c8f46dc05be20b859f00fe8168efe1e6a954fdc5cfc7246b0&scene=21#wechat_redirect](http://mp.weixin.qq.com/s?__biz=MzIxMjE5MTE1Nw==&mid=2653190796&idx=1&sn=2bf42e5783f3efd03bfb0ecd3cbbc380&chksm=8c990856bbee8140055c3429f59c8f46dc05be20b859f00fe8168efe1e6a954fdc5cfc7246b0&scene=21#wechat_redirect))

2. 本漫画纯属娱乐，还请大家尽量珍惜当下的工作，切勿模仿小灰的行为哦。



(/)

来自：程序员小灰（微信号：chengxuyuanxiaohui），作者：小灰





(/)

推荐!!!



算法与数据结构

上一篇：数据结构——二叉树的存储结构 (/article/4909.html)

下一篇：15位身份证补全为18位身份证算法 (/article/4938.html)

© 2017-2018 IT程序猿 闽ICP备08108865号-1