

题目介绍^[1]

你是一个专业的小偷，计划偷窃沿街的房屋，每间房内都藏有一定的现金。这个地方所有的房屋都 **围成一圈**，这意味着第一个房屋和最后一个房屋是紧挨着的。同时，相邻的房屋装有相互连通的防盗系统，如果两间相邻的房屋在同一晚上被小偷闯入，系统会自动报警。

给定一个代表每个房屋存放金额的非负整数数组，计算你 **在不触动警报装置的情况下**，今晚能够偷窃到的最高金额

示例 1

输入：nums = [2,3,2]

输出：3

解释：你不能先偷窃 1 号房屋（金额 = 2），然后偷窃 3 号房屋（金额 = 2），因为他们是相邻的。

示例 2

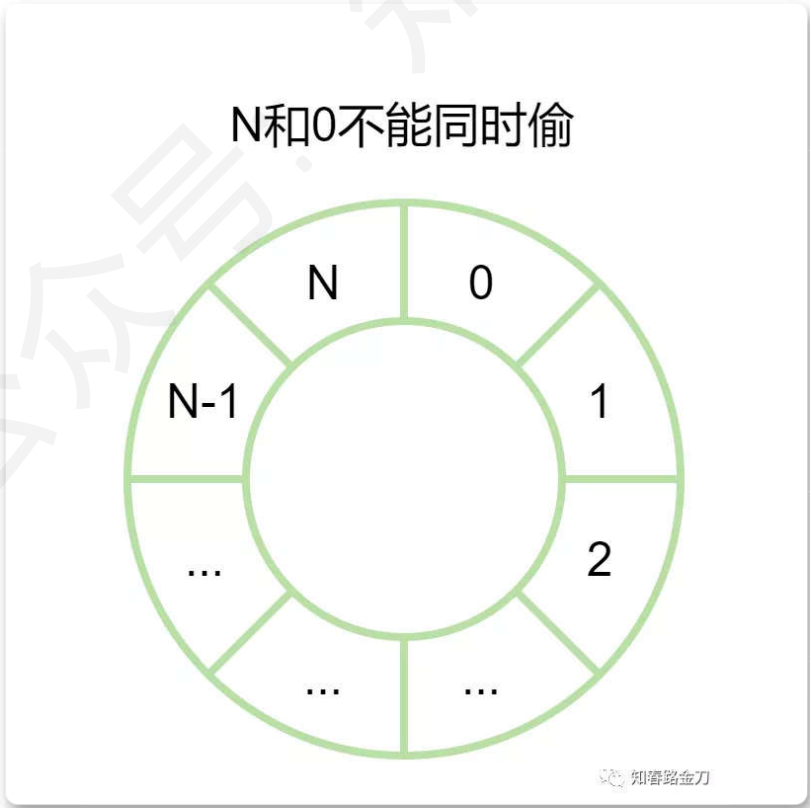
输入：nums = [1,2,3,1]

输出：4

解释：你可以先偷窃 1 号房屋（金额 = 1），然后偷窃 3 号房屋（金额 = 3）。偷窃到的最高金额 = 1 + 3 = 4。

题目解答

房子是一个环形，第一个房子与最后一个房子是相邻的，这意味着小偷如果偷了第一个房子的钱，就无法偷最后一个房子的钱，反之亦然。

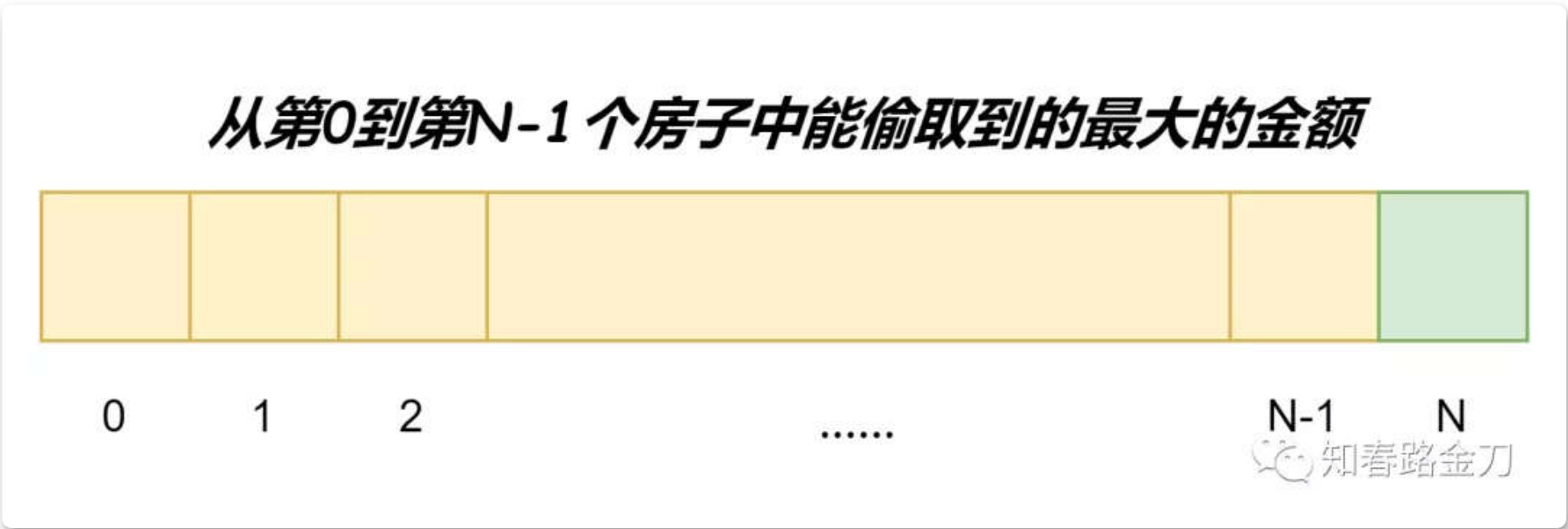


微信搜一搜

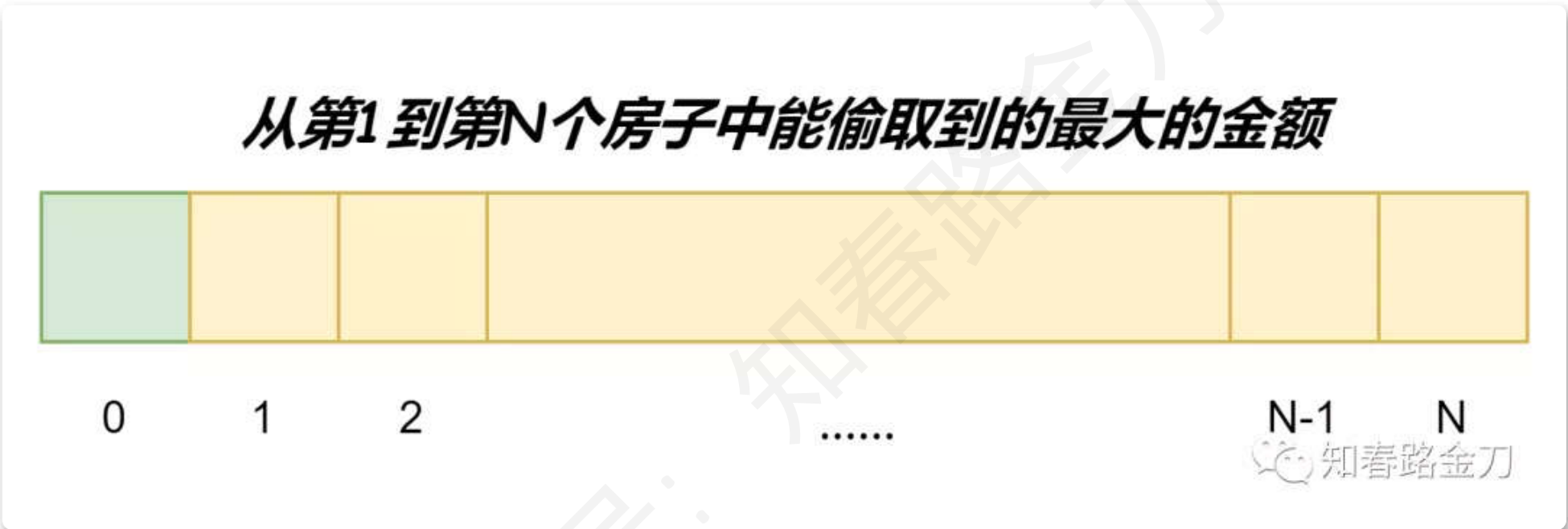
知春路金刀

如果我们对这个环进行简化，化简成一个按排进行排列的房子，那么我们只需要计算出如下两个问题：

- 从第 0 到第 N-1 个房子中能偷取到的最大的金额



- 从第 1 到第 N 个房子中能偷取到的最大的金额



取「从第 0 到第 N-1 个房子中能偷取到的最大的金额」和「从第 1 到第 N 个房子中能偷取到的最大的金额」 其中最大的一个金额，就是我们想要的答案。

上述两个问题与[打家劫舍 I](#)中的问题是一样的，如果没做过的话建议先阅读一下。下面先给出解决上面两个问题的状态转移方程，详细讲解可以看上述文章。

```
not_steal[k] = max(steal[k-1], not_steal[k-1])

steal[k] = not_steal[k-1] + nums[k]
```

其中 `steal[k]` 记录小偷偷了第 K 个房子时能获取到的最多的钱，`not_steal[k]` 记录小偷不偷第 K 个房子时能获取到的最多的钱。最后只需要取`not_steal[k]` 和`steal[k]` 中最大的一个就是我们要的答案。

方法一：动态规划



代码实现

```
class Solution {
    public int rob(int[] nums) {
        if (nums == null || nums.length == 0) {
            return 0;
        }
        if (nums.length == 1) {
            return nums[0];
        }
        if (nums.length == 2) {
            return Math.max(nums[0], nums[1]);
        }
        return Math.max(rob(nums, 0, nums.length-2), rob(nums, 1, nums.length-1));
    }
    public int rob(int[] nums, int start, int end) {
        int steal = nums[start], not_steal = 0;
        for (int i = start+1; i <= end; i++) {
            int new_steal = not_steal + nums[i];
            int new_not_steal = Math.max(steal, not_steal);
            steal = new_steal;
            not_steal = new_not_steal;
        }
        return Math.max(steal, not_steal);
    }
}
```

复杂度分析

- 时间复杂度：O(n)，其中 n 是数组长度。只需要对数组遍历一次。
- 空间复杂度：O(1)。

其他

「图解大厂面试高频算法题」专题文章主旨是：根据二八法则的原理，以付出 20%的时间成本，获得 80%的刷题的收益，让那些想进互联网大厂或心仪公司的人少走些弯路。

本专题还在持续更新 ing~ 所有文章、图解和代码全部是金刀亲手完成。内容全部放在了github^[2]和gitee^[3]方便小伙伴们阅读和调试，另外还有更多小惊喜等你发现~

如果你喜欢本篇文章，PLZ 一键三连。

参考资料

[1] 原题链接：
<https://leetcode-cn.com/problems/house-robber-ii/>

[2] github：
<https://github.com/goldknife6>

[3] gitee：
<https://gitee.com/goldknife6>

