

剑指 Offer——名企面试官精讲典型编程题（第2版）

```
// 此时更好的方法是把绳子剪成长度为2的两段，因为  $2 \times 2 > 3 \times 1$ 
if (length - timesOf3 * 3 == 1)
    timesOf3 -= 1;

int timesOf2 = (length - timesOf3 * 3) / 2;

return (int) (pow(3, timesOf3)) * (int) (pow(2, timesOf2));
}
```

接下来我们证明这种思路的正确性。首先，当  $n \geq 5$  的时候，我证明  $2(n-2) > n$  并且  $3(n-3) > n$ ，也就是说，当绳子剩下的长度大于或等于5的时候，我们就把它剪成长度为3或者2的绳子段。另外，当  $n \geq 5$  的时候， $3(n-3) \geq 2(n-2)$ ，因此我们应该尽可能地多剪长度为3的绳子段。

前面证明的前提是  $n \geq 5$ 。那么当绳子的长度为4呢？在长度为4的绳子上剪一刀，有两种可能的结果：剪成长度分别为1和3的两根绳子，或者两根长度都为2的绳子。注意到  $2 \times 2 > 1 \times 3$ ，同时  $2 \times 2 = 4$ ，也就是长度为4时其实没有必要剪，只是题目的要求是至少要剪一刀。

源代码：

本题完整的源代码：

<https://github.com/zhedahht/CodingInterviewChinese2/tree/master>

CuttingRope

### 测试用例：

- 功能测试（绳子的初始长度大于 5）。
- 边界值测试（绳子的初始长度分别为 0、1、2、3、4）。

### 本题考点：

- 考查应聘者的抽象建模能力。应聘者需要把一个具体的

```
public class B14 {  
    public static int maxProductAfterCutting(int length) {  
        /*  
         * 因为这里要求是必须剪一刀，length是绳子的长度  
         */  
        if(length < 2)  
            return 0;  
        if(length == 2)  
            return 1;  
        if(length == 3)  
            return 2;  
  
        int timeOf3 = length / 3;  
        if(length - timeOf3 * 3 == 1)  
            timeOf3 -= 1;  
  
        int timeOf2 = (length - timeOf3 * 3) / 2;  
  
        return (int)(Math.pow(3, timeOf3)) * (int)(Math.pow(2, timeOf2));  
    }  
  
    public static void main(String [] args) {  
        System.out.println(maxProductAfterCutting(8));  
    }  
}
```