

## 机器人运动范围

给你一根长度为  $n$  的绳子，请把绳子剪成整数长度的  $m$  段 ( $m$ 、 $n$  都是整数， $n > 1$  并且  $m > 1$ )，每段绳子的长度记为  $k[0], k[1] \dots k[m-1]$ 。请问  $k[0] * k[1] * \dots * k[m-1]$  可能的最大乘积是多少？例如，当绳子的长度是 8 时，我们把它剪成长度分别为 2、3、3 的三段，此时得到的最大乘积是 18。

示例 1:

输入:2 输出:1 解释:2=1+1, 1 × 1=1

示例 2:

输入:10 输出:36 解释:10=3+3+4, 3 × 3 × 4=36

```
class Solution {
public:
    int cuttingRope(int n) {
        if(n<=3)
            return n-1;
        //n>1
        //n=2 分割为 1+1 1*1=1;
        //n=3 分割为 1+2 1*2=2;
        //n>5 部分尽量分出 3 使得乘积最大
        int cout=n/3;
        if(n%3==0)
            return pow(3,cout); //可以直接分割为 3 的整数倍
        else if(n%3==1)
        {
            cout--;
            return pow(3,cout)*4; //最后一组与多余的 1 合并为 4
        }
        else
            return pow(3,cout)*2; //剩余 2 个组合一组
    }
};
```