

☑ PAT 乙级题目讲解：1001 《害死人不偿命的(3n+1)猜想》

🔗 题目简介

这是著名的卡拉兹猜想（Collatz Conjecture）问题变形版。

题目要求：对任意正整数 n ，如果是偶数就变成 $n/2$ ，如果是奇数就变成 $(3n+1)/2$ ，如此循环，直到变为 1。

目标是输出这个过程中变换的步数。

数据范围： $1 \leq n \leq 1000$

🔍 样例分析

输入：

3

我们来一步步手动模拟：

- $n = 3$ （奇数） $\rightarrow (3 \times 3 + 1) / 2 = 10 / 2 = 5$
- $n = 5$ （奇数） $\rightarrow (3 \times 5 + 1) / 2 = 16 / 2 = 8$
- $n = 8$ （偶数） $\rightarrow 8 / 2 = 4$
- $n = 4$ （偶数） $\rightarrow 4 / 2 = 2$
- $n = 2$ （偶数） $\rightarrow 2 / 2 = 1$

因此输出为：

5

共执行了 5 次变换。

🔍 解题思路

🧠 变量说明

变量名	含义
n	当前处理的正整数
cnt	记录总共的变换次数

本题的解决流程可以分为以下几个步骤：

☑ Step 1：输入处理

```
int n;  
cin >> n;
```

读取初始整数 n 。

☑ Step 2: 循环模拟变换

使用 `while(n != 1)` 作为条件，模拟过程直到 n 变为 1。

```
int cnt = 0;  
while(n != 1){  
    if(n % 2 == 0){  
        n /= 2;  
    }  
    else{  
        n = (3 * n + 1) / 2;  
    }  
    cnt++;  
}
```

其中：

- 使用奇偶性判断选择转化方式
- 每变换一步，计数器 `cnt` 加一

☑ Step 3: 输出结果

```
cout << cnt;
```

完整代码

```
#include <bits/stdc++.h>  
using namespace std;  
  
int main(){  
    int n;  
    cin >> n;  
    int cnt = 0;  
    while(n != 1){  
        if(n % 2 == 0){  
            n /= 2;  
        }  
        else{  
            n = (3 * n + 1) / 2;  
        }  
        cnt++;  
    }  
    cout << cnt;  
    return 0;  
}
```

🚩 常见错误提醒

错误类型	具体表现
忘记计数	每轮变换后没有执行 <code>cnt++</code>
奇偶判断逻辑写错	把 <code>n % 2 == 0</code> 和 <code>else</code> 写反了
条件循环不严谨	未写 <code>while(n != 1)</code> ，可能进入死循环
输出格式错误	使用 <code>endl</code> 会超时（但本题影响不大）

☑️ 总结归纳

- 学会使用 `while` 条件循环控制收敛流程
- 掌握对奇偶条件的数学建模与转换
- 理解如何使用计数器统计循环次数
- 练习边界条件处理能力（终止于 $n = 1$ ）

🧠 思维拓展

- Collatz 猜想目前尚未被证明适用于所有正整数，但本题限定数据规模小（ ≤ 1000 ），一定会收敛。
- 本题是很多更复杂模拟类题的起点，例如路径记录、变化规律建模等。