

# Euler problem 34

于船长

书山有路勤为径，学海无涯苦作舟

# 本期内容

一. 题目讲解

二. 代码演示

# 一. 题目讲解

# 一. 题目讲解

Digit factorials

145 is a curious number, as  $1! + 4! + 5! = 1 + 24 + 120 = 145$ .

Find the sum of all numbers which are equal to the sum of the factorial of their digits.

Note: as  $1! = 1$  and  $2! = 2$  are not sums they are not included.

-----

数字阶乘和

145是个有趣的数，因为 $1! + 4! + 5! = 1 + 24 + 120 = 145$ 。

找出所有各位数字的阶乘和等于其本身的数，并求它们的和。

注意：因为 $1! = 1$  和  $2! = 2$  不是和的形式，所以它们并不在讨论范围内。

# 一. 题目讲解

## 思路引导

- 1、枚举每个数字  $N$ ，判断  $N$  的每一位的阶乘之和是否等于其本身
- 2、枚举的上界是多少呢？总不能一直枚举下去，永远不退出吧。

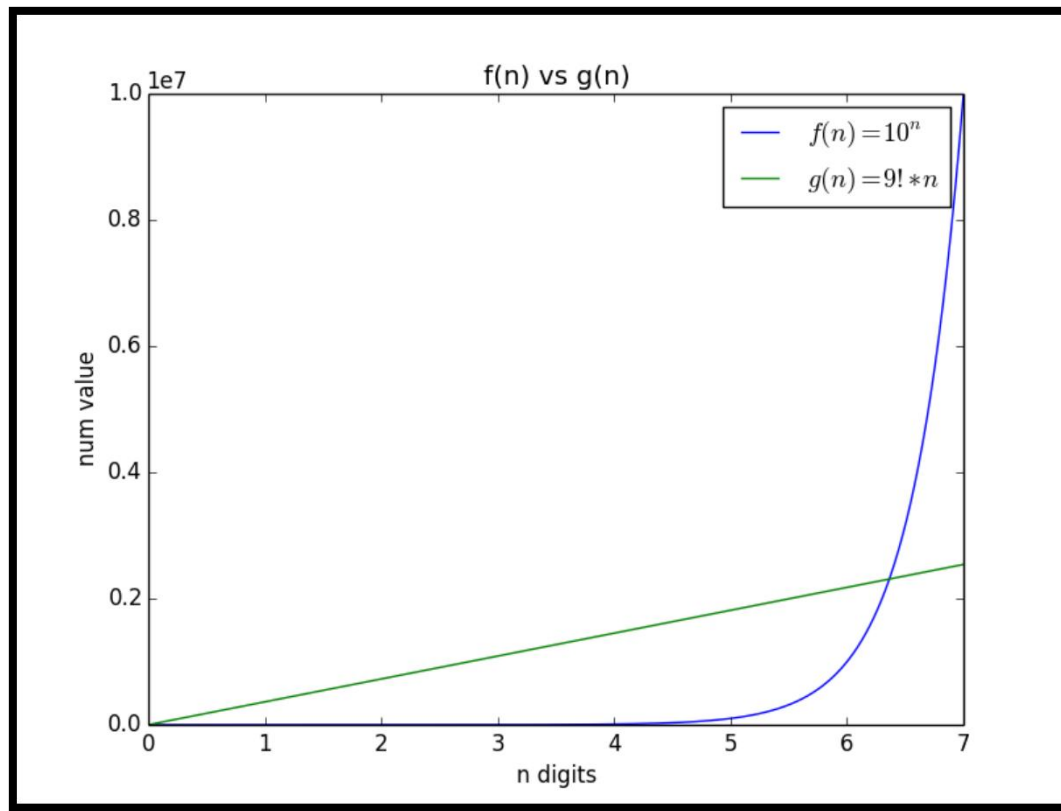
# 一. 题目讲解

估算题目中的上界，可以采用如下方法：  
设函数， $f$ ， $g$ ：

$$f(n) = 10^n$$

$$g(n) = 9! * n$$

可以发现， $f$ ， $g$ 函数单调递增，  
 $n = 6.xxxx$ 处为两个函数的交点，所以  
 $n$  取 7， $g(7)$  即为这道题目的上界。



# 一. 题目讲解

解题思路是什么？

1. 循环遍历一定范围内的每一个数字
2. 进行性质判断
3. 输出打印结果

## 二. 代码演示



## 二. 代码演示

```
#include <stdio.h>
#define max_n 2540160

int num[10];

void init(int *num) {
    num[0] = 1;
    for (int i = 1; i < 10; i++) {
        num[i] = num[i - 1] * i;
    }
    return ;
}
```

```
int is_val(int x) {
    int tmp = x, sum = 0;
    while (x) {
        sum += num[x % 10];
        x /= 10;
    }
    return sum == tmp;
}
```

```
int main() {
    int sum = 0;
    init(num);
    for (int i = 3; i < max_n; i++) {
        if (is_val(i)) sum += i;
    }
    printf("%d\n", sum);
    return 0;
}
```