

Euler problem 32

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

We shall say that an n -digit number is pandigital if it makes use of all the digits 1 to n exactly once; for example, the 5-digit number, 15234, is 1 through 5 pandigital.

The product 7254 is unusual, as the identity, $39 \times 186 = 7254$, containing multiplicand, multiplier, and product is 1 through 9 pandigital.

Find the sum of all products whose multiplicand/multiplier/product identity can be written as a 1 through 9 pandigital.

HINT: Some products can be obtained in more than one way so be sure to only include it once in your sum.

一. 题目讲解

如果一个 n 位数包含了1至 n 的所有数字恰好一次，我们称它为全数字的；例如，五位数15234就是1至5全数字的。

7254 是一个特殊的乘积，因为在等式 $39 \times 186 = 7254$ 中，被乘数、乘数和乘积恰好是1至9全数字的。

找出所有被乘数、乘数和乘积恰好是1至9全数字的乘法等式，并求出这些等式中乘积的和。

注意：有些乘积可能从多个乘法等式中得到，但在求和的时候只计算一次。

一. 题目讲解

- 1、枚举 $a*b=c$ 中的乘数与被乘数，判断(a, b, c)是否符合题目的要求
- 2、其中 $a < b$
- 3、如果 a,b,c 的位数不够9位的话，就将 b 增大一点点
- 4、如果 a,b,c 的位数超过9位的话，就 break 出枚举 b 的循环
- 5、如何快速得到 a,b,c 的总位数呢？

一. 题目讲解

函数 $f(n)$ 计算 n 是几位正整数：

$$f(n) = \lfloor \log_{10} n \rfloor + 1$$

则 a, b, c 总位数为：

$$\text{digits} = f(a) + f(b) + f(c)$$

一. 题目讲解

更一般的，求一个正整数 n 在 m 进制下的数字位数，采用如下公式：

$$g(n) = \lfloor \log_m n \rfloor + 1$$

二. 代码演示

二. 代码演示

```
01. #include <stdio.h>
02. #include <math.h>
03. #define MAX_N 100000
04.
05. int digits(int x) {
06.     return floor(log10(x)) + 1;
07. }
08.
09. int add_to_num(int x, int *num) {
10.     while (x) {
11.         if (num[x % 10]) return 0;
12.         num[x % 10] += 1;
13.         x /= 10;
14.     }
15.     return 1;
16. }
17.
18. int is_val(int a, int b, int c) {
19.     if (digits(a) + digits(b) + digits(c) - 9) return 0;
20.     int num[10] = {0};
21.     num[0] = 1;
22.     int flag = 1;
23.     flag = flag && add_to_num(a, num);
24.     flag = flag && add_to_num(b, num);
25.     flag = flag && add_to_num(c, num);
26.     return flag;
27. }
28.
29. int keep[MAX_N + 5] = {0};
30.
31. int main() {
32.     int sum = 0;
33.     for (int a = 1; a < 100; a++) {
34.         for (int b = a + 1; b < 10000; b++) {
35.             if (!is_val(a, b, a * b)) continue;
36.             if (keep[a * b]) continue;
37.             sum += a * b;
38.             keep[a * b] = 1;
39.             printf("%d * %d = %d\n", a, b, a * b);
40.         }
41.     }
42.     printf("%d\n", sum);
43.     return 0;
44. }
```