Euler problem 33

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 cancelling fractions

The fraction 49/98 is a curious fraction, as an inexperienced mathematician in attempting to simplify it may incorrectly believe that 49/98 = 4/8, which is correct, is obtained by cancelling the 9s.

We shall consider fractions like, 30/50 = 3/5, to be trivial examples.

There are exactly four non-trivial examples of this type of fraction, less than one in value, and containing two digits in the numerator and denominator.

If the product of these four fractions is given in its lowest common terms, find the value of the denominator.

一.题目讲解

消去数字的分数

49/98是一个有趣的分数，因为缺乏经验的数学家可能在约简时错误地认为，等式 49/98＝4/8之所以成立，是因为在分数线上下同时消去了9的缘故。

显然，存在诸如30/50＝3/5,这样的平凡解。

除此之外，在所有值小于1且分子和分母都是两位数的分数中，恰好有四个非平凡解。 将这四个分数的乘积写成最简分数，并求此时分母的值。

一.题目讲解

1. 如何判断两个分数相等，转换成乘法式子进行判断，是编程中经常用的技巧

2. 所谓最简分式，指分子分母的最大公约数为1

3. 其余事情，按照题目意思编程实现即可

二. 代码演示

二. 代码演示

```c
#include <stdio.h>

int gcd(int a, int b) {
    return (b ? gcd(b, a % b) : a);
}

int is_val(int x, int y) {
    int x1 = x / 10, x2 = x % 10;
    int y1 = y / 10, y2 = y % 10;
    if (x2 == 0 || y2 == 0) return 0;
    if (x1 == y1 && x2 * y == y2 * x) return 1;
    if (x1 == y2 && x2 * y == y1 * x) return 1;
    if (x2 == y1 && x1 * y == y2 * x) return 1;
    if (x2 == y2 && x1 * y == y1 * x) return 1;
    return 0;
}

int main() {
    int x = 1, y = 1, c = 1;
    for (int a = 10; a < 100; a++) {
        for (int b = a + 1; b < 100; b++) {
            if (!is_val(a, b)) continue;
            x *= a;
            y *= b;
            c = gcd(x, y);
            x /= c;
            y /= c;
        }
    }
    printf("%d\n", y);
    return 0;
}
```