

P3951 解题报告

【题目描述】

[P3951_problem](#)

【题目分析】

- 审题可知本题要求求出一个最大的，当前持有两种面值金币无法支付的价格。
- 最大无法支付价格实质上是 $ax + by = c$ 中使得 x, y 无非负整数解的 c 的取值的最大值，思路方向应偏向扩展欧几里得算法。
- 可以考虑上暴力，可以得到 30 分的好成绩（确信）。
- 根据输入数据的特殊性质： a, b 互质，可以尝试用特殊算法解决。

【解题过程】

• 先考虑暴力代码，将 a, b 的系数分别在 $[0, 10000]$ 区间内枚举，并将可以用这两种面值金币凑出的价格标记，最后再从标记数组尾部向头部扫描，找到最大的未标记数并输出。暴力代码如下：

```
17 for(int i=0;i<=10000;i++)
18     for(int j=0;j<=10000;j++)
19         if(!p[i*a+j*b]) p[i*a+j*b]=true;
20 for(int i=1;i<=50000;i++)
21     if(!p[i]) ans=i;
```

• 回到题目中发现，本题要求我们构造一个最大不合法数，那么由其定义可以知道这个最大不合法数加上 1 必然为一个合法数。所以此时任务被转化为“构造一个最大的，前一个位不合法数的合法数 k ”。由于 k 是一个合法数，所以其必然满足 $ax + by = k, x \in \mathbb{N}, y \in \mathbb{N}, k \in \mathbb{N}$ 。若 $k-1$ 也为合法数，则 $k-1$ 必然满足以下两个式子之一：

$$\begin{cases} a(x - x_1) + b(y + y_1) = k - 1 \\ a(x + x_2) + b(y - y_2) = k - 1 \end{cases}$$

• 同理，要使 $k-1$ 为非法数，那么只需要使 $x - x_1 < 0$ 或 $y - y_2 < 0$ 即可，其中， x_1, y_1 为 $ax + by = 1$ 中 x 为最小非负整数的解， x_2, y_2 为 $ax + by = 1$ 中 y 为最小非负整数的解。那么最大的并且前一个数为不合法数的合法数就是 $a(x_1 - 1) + b(y_2 - 1)$ ，最大的非法数即为 $a(x_1 - 1) + b(y_2 - 1) - 1$ 。因为对 x_1, y_1, x_2, y_2 的定义，故无法找到比 $a(x_1 - 1) + b(y_2 - 1)$ 更大的前一个数为非法数的合法数。综上，最大非法数即为 $a(x_1 - 1) + b(y_2 - 1) - 1$ 。因此，通过对 `exgcd` 的写法稍作改动，求出 x_1, y_2 的值即可得出正解。代码如下：

```
16 exgcd(a,b,x,y);
17 if(x>0)
18 {
19     swap(a,b);
20     swap(x,y);
21 }
22 long long t=-x/b;
23 x+=(t*b);
24 y-=(t*a);
25 while(x<0)
26 {
27     x+=b;
28     y-=a;
29 }
30 while(x>0)
31 {
32     x-=b;
33     y+=a;
34 }
35 ans=a*(x+b-1)+b*(y-1)-1;
```

•通过继续分析，我们发现本题算法的时间复杂度可以进一步优化到 $O(1)$ 。由我们之前写出的暴力枚举代码，将a,b分别取互质随机数打表如下：

x	y	ans
3	7	11
5	7	23
17	5	63
19	5	71
19	13	215
23	29	615

•经过不完全归纳我们可以初步得出一个猜想：最大非法数ans满足 $ans = ab - (a + b)$ 。
 •得出猜想，我们现在需要求证这个结论。由于本题的操作都是有关数字组合求和的，所以我们先在 Excel 中打表找出规律。

•我们将所有数根据与a取模的结果分为a类，相应地，我们在数轴上从原点向正方向截出长度为a的无数段，并将所有的段归纳为一个区间。根据最初得到的方程 $ax + by = k$ ，我们可以推导得到x的值与k对a取模的值无关，而由y决定。换言之，我们在之前的程序中枚举x,y来凑钱数的操作本质上就是在这个区间中对方格进行覆盖。例如在下图中，第一列第三行中， 1×3 将模7结果为3的方格全部覆盖，意味着任何在 1×3 之后对7取模得3的数都是合法数，以此类推。不难发现要完成整个区间的覆盖需要 $(7 - 1)$ 次操作，即最后一次覆盖时3的系数为6，而此时若从凑出的钱数中取出7得到11，则11这个对7取模得4的数是非法数，因为满足对7取模得4的合法数是在 3×4 后才会被覆盖到，即必须大于或等于12，而此时11的在数轴上的位置为 $(7 - 1) \times 3 - 7$ 。

3的系数 模7结果	1	2	3	4	5	6
1					5×3	
2			3×3			
3	1×3					
4						6×3
5				4×3		
6		2×3				
0						

•由上述推导过程就可以得到最大非法数 $ans = a(b - 1) - b$ ，将其化简后得到本题的通项公式： $ans = ab - a - b$ 。代码如下：

```

14 scanf("%lld%lld",&a,&b);
15 ans=a*b-a-b;
16 printf("%lld",ans);

```

•在漫长的抄题解写题解过程中，笔者偶然发现，由于某些玄学因素，在 exgcd 写法中，即便不运行扩展欧几里得算法的子函数，只要在主函数中最后的结果中加上1即为正解，原因不明，代码如下：

```

13 int main()
14 {
15     scanf("%lld%lld",&a,&b);
16     exgcd(a,b,x,y);
17     if(x>0)
18     {
19         swap(a,b);
20         swap(x,y);
21     }
22     long long t=-x/b;
23     x+=(t*b);
24     y+=(t*a);
25     while(x<0)
26     {
27         x+=b;
28         y+=a;
29     }
30     while(x>0)
31     {
32         x-=b;
33         y-=a;
34     }
35     ans=a*(x+b-1)+b*(y-1);
36     printf("%lld",ans);
37     return 0;
38 }
39 void exgcd(long long a,long long b,long long &x,long long &y)
40 {
41     //笔者也不知道为什么这里不运行也能AC
42 }

```

【完整代码】

[P3951_violence.cpp](#)

[P3951_final_exgcd.cpp](#)

[P3951_final_0\(1\).cpp](#)

[P3951_final_magic.cpp](#)